FINAL PROJECT REPORT

| | |
|---|---|
| Project Team Lead | **UIUC** |
| Project Title | **Capability Modeling for Digital Factories (CaMDiF)** |
| Project Designation | **15-12-05** |
| UI LABS Contract Number | **0220160024** |
| Project Participants | **Texas State University**<br>**Innovation Machine**<br>**ITAMCO** |
| DMDII Funding Value | **$147,208.00** |
| Project Team Cost Share | **$144,634.00** |
| Award Date | **12/22/2016** |
| Completion Date | **4/30/2018** |

SPONSORSHIP DISCLAIMER STATEMENT: This project was completed under the Cooperative Agreement W31P4Q-14-2-0001, between Army Contracting Command - Redstone and UI LABS on behalf of the Digital Manufacturing and Design Innovation Institute. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of the Army.

DISTRIBUTION STATEMENT A. Approved for public release.

# Table of Contents

# 1 Executive summary:

## 1.1 High Level Problem

This project is motivated by the need for improving the agility and intelligence of supplier discovery and evaluation solutions and also enhancing the visibility of SMEs in the cyber-space. The main technical problem addressed in this project is formal and standard representation and analysis of manufacturing capabilities.

The manufacturing industry is undergoing profound changes brought about by the emergence of service-oriented, cloud-based, and digital manufacturing paradigms. The democratization of manufacturing is among the most visible trends that have reshaped the manufacturing landscape within the past few years. With a lowered barrier to entry, a larger number of small-to-medium sized enterprises (SMEs) are capable of offering diverse manufacturing services both internally and externally through building virtual supply networks and exploiting the resources provided by distributed partners. Consumers of manufacturing services can benefit from a larger and more diverse supply pool since they are provided with a wider range of options when searching for qualified suppliers. Nevertheless, the sheer size of the supply pool presents multiple challenges to efficiently evaluating and selecting manufacturing suppliers.

Traditional approaches to manufacturing capability evaluation and supplier selection often entail direct interaction with the supplier and possibly visiting the supplier's facility to obtain better insight into the technological and organizational capabilities of the supplier. Supplier visits, accompanied by pilot production runs will result in accurate evaluation of suppliers' capabilities but this approach is not scalable when it comes to evaluating of large groups of suppliers in an agile business environment. As the interaction between suppliers and customers becomes increasingly virtual and the lifespan of supply chains becomes shorter, more efficient and intelligent approaches to capability evaluation are called for. Virtual capability analysis can be conducted through web search, online surveys, or exploring the profiles of suppliers on e-sourcing portals but they only provide rudimentary information about suppliers' capabilities. This issue is compounded by the lack of structure and formality is the way both suppliers and e-sourcing portals represent and advertise manufacturing capabilities. There are no industry standard information models for capability representation and modeling. This is major gap that needs to be filled in the digital manufacturing ecosystem.

## 1.2 High Level Purpose

The high level purpose of this project is to develop a framework for manufacturing capability representation and dissemination to enable agile supply chain formation. To serve this purpose, a formal reference ontology for representation of manufacturing capabilities of contract manufacturing companies, with specific focus on SMEs, is developed and validated experimentally. The hypothesis is that with a reference ontology, manufacturing suppliers can describe their manufacturing capabilities more accurately and comprehensively. Using a standardized terminology enhances information interoperability throughout the lifecycle of a supply chain. As a result, supply and demand entities can be matched to each other with more precision. Also, since the ontology uses machine-understandable semantics, the automation of supplier evaluation and supply chain formation processes can be realized more efficiently.

## 1.3 Summary of Findings and Recommendations

One of the main findings of this project was that accurate and standard representation of manufacturing capability is a very critical need for both OEMs and SMEs. OEMs need more rigorous models and method for evaluating prospective suppliers in terms of their technological capabilities. SMEs need to enhance their visibility through advertising their capabilities in an accurate and verifiable manner. However, the problem of capability representation is not addressed adequately in practice. Capability representation is often conducted in an ad hoc manner using informal and incomplete templates and vocabularies. There isn't even a universally agreed-upon definition for manufacturing capability and its sub-types. It is necessary for the manufacturing industry in general, and the contract manufacturing industry in particular, to take a more systematic and holistic approach to representing and sharing manufacturing capabilities and skills. Using open-source reference ontologies for unifying the semantics of this highly heterogeneous domain as can significantly reduce the deficiencies caused by proprietary and incompatible information models. This work sets the stage for a developing an open-source platform for capability modeling using semantic web technology. The future goal is to build a community of users and developers around the developed platform in order to extend the ontology in a collaborative fashion, supported by the necessary governance mechanisms, and to promote the adoption of formal capability models in industry through demonstrating multiple use cases.

# 2 Project Overview:

In this section, the overview of the developed technology framework is provided. The developed framework is referred to as CaMDiF (Capability Modeling for Digital Factories). For the purpose of this work, *Digital Factory* is defined as the digital twin of a physical production facility, supplemented by the ontological representation of the facility that describes the facility in terms of installed machinery, human skills, and other production support systems and resources, including both hardware and software.

## 2.1 Project Scope and Objectives

The *business problem* that will be solved by the developed technology solution is rapid deployment and customization of agile supply chains in virtual environments. The **objective** of this project is to significantly increase the intelligence and effectiveness of various supply chain decisions including *sourcing* and *capability and capacity adjustment* through:

- Developing an ontology for manufacturing capability representation to enable semantic interoperability and structured information exchange throughout the supply chain. The ontology is generic and extensible enough to cover a wide spectrum of manufacturing processes.
- Providing SMEs with highly visual, user-friendly, and intuitive user interfaces for creating the digital twin of their facilities and sharing their formal capability models using the developed ontology.
- Developing the Proof-of-Concept software framework for factory digitization, capability analysis, and supply chain configuration.
- Providing real-time, dynamic insight into the technological capabilities, capacities, and quality history of prospective suppliers through sharing their formal capability models.
- Automating the sourcing process by enabling active participation of software agents in sourcing decisions.
- Improving the cyber-visibility of manufacturing companies.

The **scope** of this project is limited to SMEs in contract manufacturing industry. With respect to manufacturing processes, this project is focused on CNC machining and Additive Manufacturing. These processes were selected for the following reasons:

- The high number of SMM (Small to Medium Size Manufacturers) that perform these processes
- The most widely used and emerging manufacturing process in the U.S.
- High demand from large manufacturers, especially DoD contractors
- Expertise of team members; MSDL ontology maturity

## 2.2 Technical Approach

The following principles summarize the technical approach taken in this project:

### 2.2.1 Open-source principle:

The CaMDiF framework and its ontology are developed based on *open-source* and *web-native* standards and protocols. The interfaces of the software framework are designed to hide the complexities of the underlying ontology. The ontology of CamDiF is based on the Web Ontology Language (OWL). OWL is the ontology language of semantic web recommended by World Wide Web Consortium (W3C). Also, the ontology is aligned with Basic Formal Ontology (BFO) as a generic upper-level ontology. The ontology of CaMDiF is based on Manufacturing Service Description Language (MSDL). The digital factory generated using CamDiF can be reused by third-party application due to its open syntax and semantics.

### 2.2.2 Inference Principle:

Since capabilities arise from resources, the inference logics of the framework are formulated such that the capability and service models of the digital factory can be inferred automatically from the available resources.

## 2.3 Planned Benefits:

The explicit and implicit capability and service models can be published using the open-source and standard ontology to be used by third-party applications for different purposes such as supplier selection and manufacturability analysis. This can enhance the visibility of small and medium-sized manufacturers in the virtual space. Manufacturing companies can share and publish their factory service models in order to advertise their capabilities in centralized or decentralized manufacturing marketplaces and service-oriented platforms. Through exploring and querying the capability and service models of digital factories, companies can develop a deeper and more precise understanding of the technological capabilities of prospective suppliers, thus making more informed decisions when building supply chains. The benefits of CaMDiF framework for manufacturing suppliers and OEMs are listed in Table 1 and Table 2.

**Table 1: The benefits of CaMDiF framework for manufacturing suppliers**

| CaMDiF framework enables manufacturing suppliers to: |
|---|
| • Describe their technological capabilities in terms of manufacturing services in a machine-readable fashion using an open-source standard<br>• Describe the parts produced in the past and the qualities achieved<br>• Create a “digital twin” of their facility through selecting their installed equipment and machines from a given library of physical resources (*Drag & Drop Factory*)<br>• Update their capability model in real-time through updating the configuration and layout of the digital factory<br>• Find the right customers through using the automated matching utility provided by the platform<br>• Evaluate their technological readiness and competencies based on the current demand through using the capability scoring utility provided by the platform |

**Table 2: The benefits of CaMDiF framework for OEMs**

| CaMDiF framework enables manufacturing OEMs to: |
|---|
| • Evaluate the technological capabilities of prospective suppliers through capability visualization and scoring utilities.<br>• Find the right suppliers through using the automated supplier search and evaluation tool.<br>• Deploy supply chains rapidly using the service composition and orchestration utility provided by the platform.<br>• Mitigate their risks through on-demand consumption of the pooled manufacturing capacities and capabilities available on the cloud. |

## 2.4 Metrics Analysis & ROI Assessment

This section provides and overview of CaMDiF framework through describing its system requirements, architecture, and its main modules and features. Since the core component of CaMDiF framework is its ontology, more in-depth discussion of the ontology is provided.

## 2.5 Technology outcomes

### 2.5.1 System Overview

The developed technology solution (CaMDiF framework) enables users to extract the capabilities of a given manufacturing facility using intuitive and visual user interfaces. Libraries of resources (CNC machines and 3D printers) provide the users with a wide range of equipment to choose from (Figure 1). Third-party apps can use the capability and service models created in CaMDiF for a variety of purposes such as design for manufacturability (DFM) and supply chain planning (Figure 2).

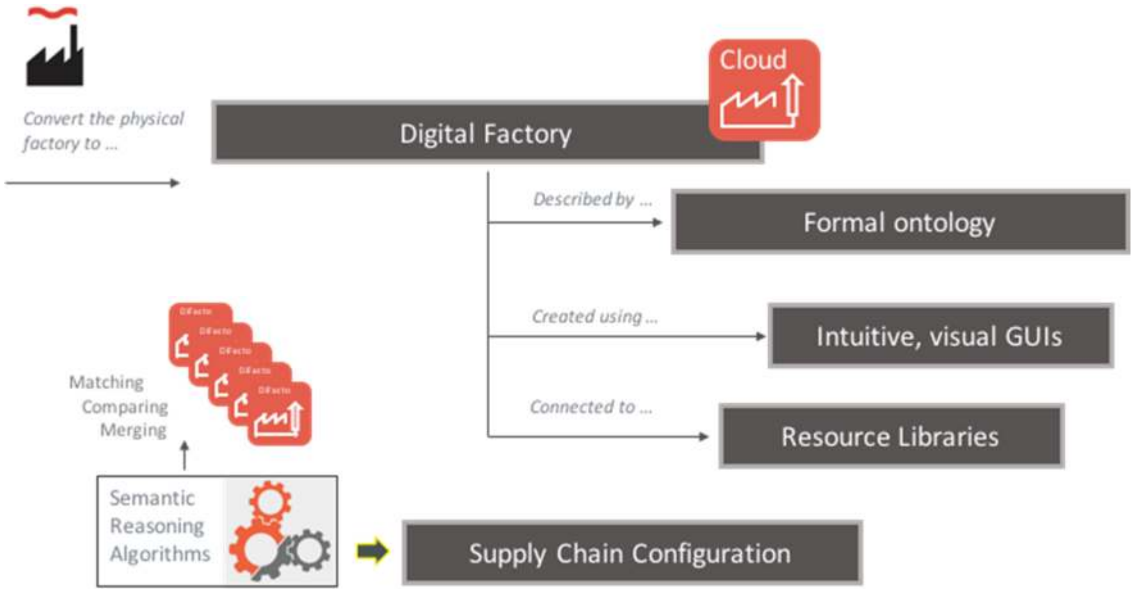



**Figure 1: CaMDiF enables factory digitization and supply chain configuration**

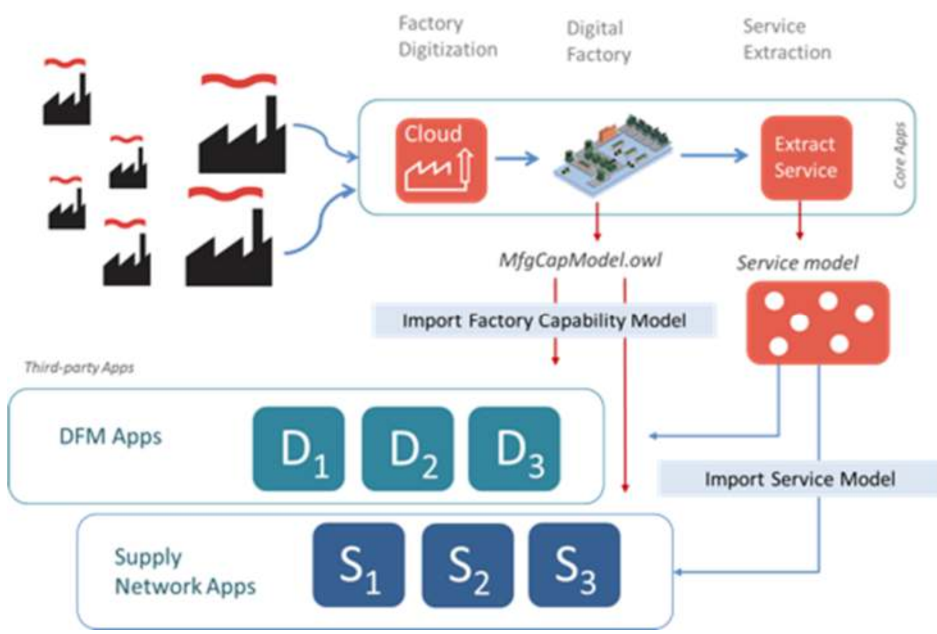



**Figure 2: 3rd party apps can use the capability models generated by CaMDiF platform**

### 2.5.2 System Requirements

The main requirements of the system are listed below:

- CaMDiF is easy to use and it is affordable
- CaMDiF hides the complexities of its underlying ontology and information model
- CaMDiF is a web-native solution
- CaMDiF uses open-source architecture

- CaMDiF used accepted standards in information modeling and knowledge representation.

### 2.5.3 System Architecture

The CaMDiF framework has a three-level architecture as shown in Figure 3.The main components of the *data and knowledge layer* are the MSDL ontology, the manufacturing capability thesaurus, external domain ontologies, and the libraries of manufacturing resources including CNC machine and 3D printer, factory, and supply chain libraries. The second layer is the *semantic layer* which is basically the Apache Jena semantic application suite which provides a set of Java libraries and Application Programming Interfaces (APIs). Jena allows programmers to create, edit, and manage semantic web ontologies using RDF graphs. Also, Jena provides the necessary interfaces for query and reasoning that are usually needed in semantic applications. The last layer is *the application layer* that has three main functions, namely, *build*, *analyze*, and *match*.

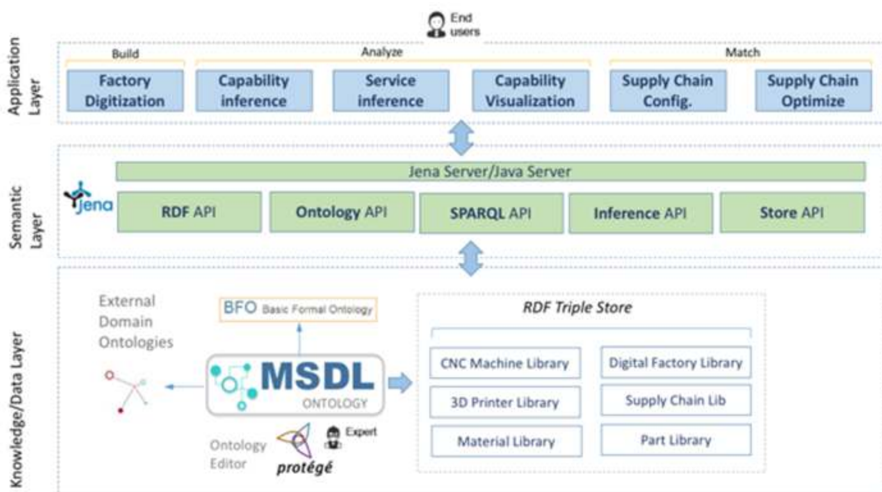



**Figure 3: The system architecture of the CaMDiF framework**

### 2.5.4 Features & Attributes

CaMDiF framework is composed of the three main modules, namely, build, analyze, and match.

#### *2.5.4.1 Build module:*

The Build module provides a set of functions and libraries required for creating the digital twin of a manufacturing facility and inferring its capabilities. Using the factory digitization function within the build module, manufacturing companies can interactively create the digital model of their facility and annotate it with explicit capability and capacity-related information. The interfaces hide the complexities of the underlying knowledge models used in the framework's knowledge-base. Also, simple user interfaces encourage rapid and regular update of the digital factory such that it accurately mirrors the physical facility. The digital factory is connected to a library of manufacturing resources, allowing the user to populate the factory model with the right set of resources. Any change in the digital facility will be reflected in the supplier's service and capability models in real-time.

#### *2.5.4.2 Analyze module:*

The *capability inference* and *service inference* functions under the Analyze Module are used for automatically inferring the capabilities and services associated with the digital factory. It uses the explicit capability information provided by the supplier and expands upon it through discovering latent and implicit capability patterns. The extracted manufacturing capability is represented ontologically. Capability extraction is a knowledge-intensive process that capitalizes on the domain knowledge already encoded in the ontology. Capability extraction is a bottom-up process starting with the device and machine-level capability model going up to the supply chain level. The capability extraction module can be used, by third-party applications, to create regional models of manufacturing capability and represent them through "capability heat map" thanks to the rigorous capability quantification algorithms embedded in this module.

#### *2.5.4.3 Match module:*

The Match Module provides the functionalities required for matching the production work orders with the factories that have the required capabilities to fulfill the order. In the CaMDiF framework, supply and demand entities are translated into units of manufacturing service with well-defined capability expectations. Therefore, semantic matchmaking between requested services and provided services is translated into matching between requested and provided capabilities.

### 2.5.5 Capability Ontology

To formally define the capabilities of the digital factory, MSDL (Manufacturing Service Description Language) [7] is used as the underlying ontology. MSDL was extended and modified to meet the needs of CaMDiF framework. MSDL was also aligned with the Basic Formal Ontology (BFO) which has resulted in a significant change in the class structure of the ontology. More details on alignment with BFO is provided in section 10.2. Throughout this paper, the term CaMDiF always refers to the framework and not the ontology.

#### *2.5.5.1 MSDL*

MSDL is a descriptive ontology, based on Web Ontology Language (OWL), that was developed for representation of capabilities of manufacturing services. It was originally developed by Farhad Ameri at the University of Michigan and then maintained and extend at Texas State University. MSDL decomposes the manufacturing capability into four levels of abstraction, namely, supplier-level, shop-level, machine-level, and device-level. The capabilities of every instance of the Digital Factory is formally described using the MSDL ontology. A unique feature of MSDL is that it is built around a *service-oriented paradigm*, therefore, it can be used for representing a manufacturing system as a collection of manufacturing services with specific capabilities. MSDL was initially designed to enable automated supplier discovery in distributed environments with focus on mechanical machining services. However, to address the needs of the CaMDiF project with respect to capability modeling, multiple classes were added to MSDL to represent different types of technological and organizational capabilities. MSDL has a wide range of classes but in this paper, the ones that are directly related to the notion of Digital Factory are presented and discussed.

Some of the MSDL classes, relevant to CaMDiF project, are defined below:

- *Manufacturing Company*: A business entity involved in production of goods.
- *Factory*: A collection of production machines and other supporting equipment used to make large quantities of goods.
- *Facility*: The building with its facility systems together with all production equipment inside the building.
- *Manufacturing Capability*: The abilities of a production entity related to fabricating a unit of product.
- Pr*oduction Capability*: The abilities of a production entity related to production of large volumes of products.
- *Service*: intangible product that is instantly consumed as it is produced
- *Manufacturing Process*: a function of a production equipment that results in change in the geometric and/or mechanical properties of the input entities.

Figure 4 shows the class diagram related to different types of capability for a production machine. Figure 5 shows different types of production and manufacturing capabilities for a typical factory.

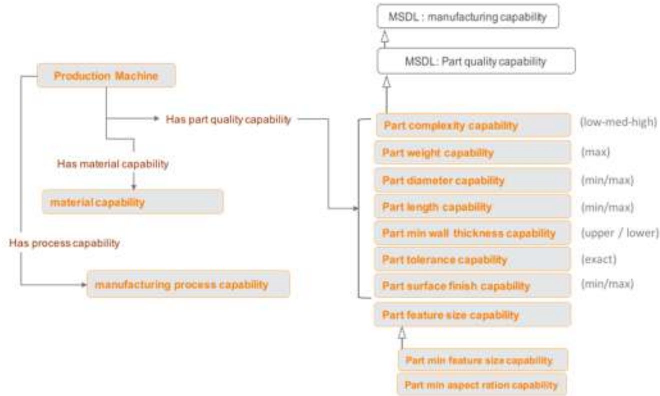



**Figure 4: Different types of capability for a production machine.**

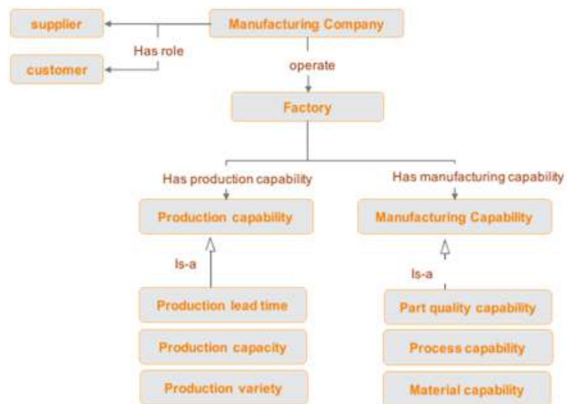



**Figure 5: subclasses of production capability and manufacturing capability classes**

### 2.5.5.2 Alignment of MSDL with BFO

Since MSDL is designed for interoperability, it is imperative to align it with an upper ontology so that it can be easily integrated with other manufacturing ontologies in a hub-and-spoke architecture. The modified version of MSDL that is implemented in the CaMDiF framework uses Basic Formal Ontology (BFO) as the foundational, or upper, ontology [8]. BFO is deliberately designed to be very small and its most recent version, BFO 2.0, has 35 classes. As a domain-neutral upper-level ontology, BFO adopts a view of reality and represents different types of entities that exist in the world and relations between them. The notion of *ontological realism* amounts to the idea that an ontology should be analogous not to a data model, but rather to a reality model [9]. This maximizes the utility and stability of the ontologies that are based on BFO.

BFO can be used as an integration hub for domain-specific ontologies. BFO is particularly used widely in the biomedical and biological domain [10]. There are two types of entities in BFO, namely, *continuants* and *occurrents*. Continuants are the entities that continue to persist through time while maintaining their identity whereas, occurrents are the events or happenings in which continuants participate. Apart from its realistic approach, BFO has multiple other unique features that make it an appropriate upper ontology for many domains. Firstly, BFO has a very large user base and it is widely used in a variety of ontologies. Secondly, BFO is very small and correspondingly easy to use and easy to learn. Additionally, BFO is very well-documented and there are multiple tutorials, guidelines, and web forums for using BFO in ontological projects.

Since one of the major contribution of this work is integrating MSDL with BFO, the next section describes how various classes of MSDL are mapped to BFO classes.

#### 2.5.5.3 MSDL Continuants

There are three types of continuants in BFO, namely, generically dependent continuants, independent continuants, and specifically dependent continuants.

Generically dependent continuants: a generically dependent continuant is a continuant that is dependent on another continuant as its bearer. It can migrate from one bearer to another such as a pdf file that can exist on multiple flash memories. *Information content entity* is a type of generically dependent continuant that is *about* another entity. For example, the measured value of the length of a shaft is a piece of information about the shaft. A measured value that is the recording of the output of a measurement process is an instance of '*measurement datum'* class. Most of the generically dependent classes in MSDL are subclasses of '*measurement datum'* class. Each measurement datum has a value and a unit label. Instances of measurement datum play a pivotal role in quantifying the capabilities of manufacturing systems, including machines, machine cells, and factories.

Independent continuant: Independent continuants do not depend on other entities for their existence. For example, a machine tool or a 3D printer can exist independently as a standalone entity. The sub-categories of independent continuant class in BFO are *material entity* and *immaterial entity*. An instance of material entity is a continuant that includes some portion of matter. *Object* is a sub-class of material entity. Examples of object in MSDL are dies, fixtures, cutting tools, workpieces, engineered artifacts, etc. Parts of production equipment such as machine spindle and machine table are also considered to be objects. A CNC machine is an *object aggregate* in BFO since it is composed of multiple parts that are objects themselves. Figure 6 shows the components of an instance an Okuma vertical mill.

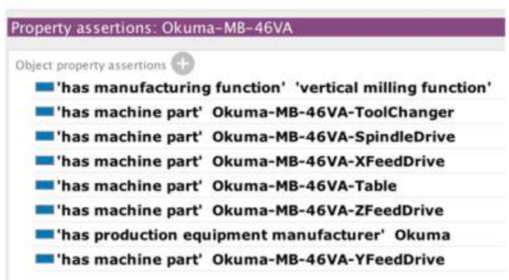



**Figure 6: parts of Okuma vertical mill**

Examples of immaterial entities in MSDL include one-dimensional boundaries such as X axis of a machine tool or three-dimensional sites such as the interior of the build chamber of a 3D printer or the working envelope of a CNC vertical mill.

Specifically dependent continuant: Specifically dependent continuants, such as color or shape, cannot migrate from one bearer to another and they depend on a specific bearer such as this machine in this machine shop or this person in this room. *Quality* and *realizable entity* are two major subcategories of specifically dependent continuant in BFO.

*Quality* is a specifically dependent continuant that is exhibited or manifested only if it is inhered in an entity. Examples of quality include the mass of a work holder, the shape of a printed part, the temperature of a machine coolant liquid, or the length of a machine table. Some of the physical qualities in MSDL are shown in Figure 7. Figure 8 shows some of the physical and temporal qualities of an instance of a tool changer system. For example, the time it takes to remove a tool from a part and bring the next tool to the part (i.e., chip to chip time), is a temporal quality of the tool changer.

*Realizable entities* require some type of process through which they can be realized. There are two sub-categories of realizable entities in BFO, namely, function and role.

A Role exists only because its bearer is in a special set of social, physical, or institutional circumstances. For example, the role of a person as a manufacturing engineer can be realized when the person is involved in a set of activities related to manufacturing engineering profession. Also, a manufacturing company can play different roles such as the role of a supplier or the role of a customer depending on the circumstances. Roles are externally grounded realizable entities since they are awarded to the bearer of the role by external agents. A milling machine can play the role of a backup machine for repair purpose only and this role can be assigned to the machine by the plant manager.

*Functions*, on the other hand, are internally grounded dispositions since their realization depends on the physical makeup of their bearer. The function of a milling machine is removing material. A milling machine can deliver this function because it is equipped with the right set of systems and tools required for removing material through the searing process. But the degree to which a milling machine can create smooth surfaces is interpreted as the *capability* of the milling machine and not its function. Capability is not a BFO class yet. Therefore, it was defined in MSDL as a subclass of *bfo disposition*.

Figure 9 shows the class diagram for surface roughness capability class in MSDL. As mentioned before, capability is a measurable entity. Surface roughness is a capability measured as a length measurement datum. Two identical milling machines might have different surface finish or tolerance capabilities because they are maintained differently. Some of the MSDL continuants are shown as example in Figure 10.


**Figure 7: Physical qualities in MSDL**

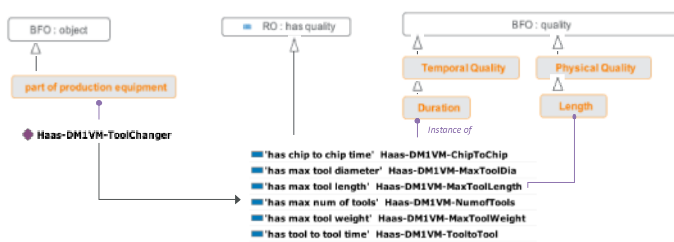



**Figure 8: Physical and Temporal qualities of tool changer system**

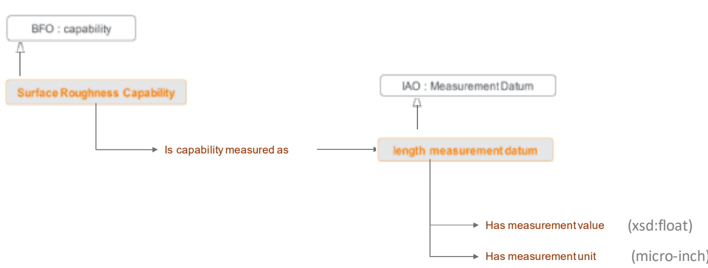



**Figure 9: class diagram for surface roughness capability**

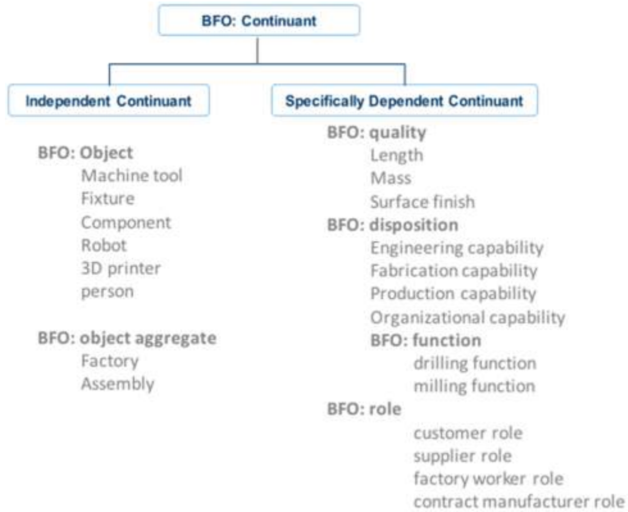



**Figure 10: some examples of continuants in MSDL**

#### 2.5.5.4 MSDL Occurrents

In BFO, occurrents are the events or happenings that unfold themselves in time. *Process* class is one of the main subcategories of occurrent class. Two MSDL classes that are sub-classes of BFO:process are *manufacturing process* and *service*. Manufacturing process is a process, enabled by some equipment, which alters the shape and/or properties of the input material. It should be noted that manufacturing *process* is different form manufacturing *function* since the former is an occurrent while the latter is a continuant. The function of a drilling machine has no temporal parts but a specific drilling process (or operation) should take place in a certain time interval, hence being an occurrent.

### 2.5.6 Capability Inference

Using the capability ontology, one can create a formal representation of a factory's capability model. From the capabilities explicitly represented in the model, new capabilities can be inferred automatically using the reasoning services provided by the MSDL ontology. In this section the approaches used for inferring manufacturing capabilities are discussed. It should be noted that such inferences at best provide an approximation of the latent capabilities of the factory and some level of uncertainty is always assumed when inferring new capabilities. Four categories of capability are discussed in this section: 1) part quality capability, 2) process capability, 3) material capability, and 4) production capability.

#### 2.5.6.1 Part Quality Capability Inference

A machine tool can create certain qualities such as tolerance, surface roughness, or minimum feature size on a part. The range of these qualities define the capability of the machine tool. The collective capability of the factory is calculated through aggregating the capabilities of individual machines in the factory.

#### 2.5.6.2 Surface finish capability:

Figure 11 shows the procedure for calculating the surface finish capability of a factory. According to this procedure, the surface finish capability value for each factory machine, already stored as instance information, is retrieved. If the retrieved value is null, then the immediate superclass of the machine tool is queried instead and the surface finish capability value is retrieved. The reasoning behind this approach is that the parent machine can provide a reasonable approximation of the capabilities of the children machines. If none of the higher-level individuals can provide a value for surface finish capability, then a *generic machine* from the same machine vendor is used as the reference machine to provide some approximation about the capability of the machine. The generic machine from a given vendor is the average machine with respect to capabilities based on the vendor's product portfolio.

FOR i=1 to num of machines in the factory f [instance of MSDL: factory]

$m_i$ [instance of MSDL: machine tool]

Retrieve part surface finish capability value $m_{i}\_ sf_{cap}$

```
IF mi- sfcap = Null
    Find an instance of the superclass of => smi
Retrieve part surface finish capability value smi_ sfcap
    IF smi _ sfcap = null  THEN
    Find the instance of a Generic Machine from the same vendor => gmi
        Retrieve part surface finish capability value gmi _ sfcap
                IF gmi- sfcap = Null  THEN Let mi- sfcap = null
ELSE  Let mi- sfcap = gmi- sfcap  AND go to the next machine End IF
        ELSE Let mi- sfcap = smi _ sfcap AND go to the next machine
        End IF
End IF
Factory - sfcap =Min (mi- sfcap) for all values of i
```

**Figure 11: The procedure of calculating the surface finish capability of a factory**

This procedure is based on the simplifying assumption that surface finish capability is a standalone capability. However, more realistically, surface finish capability is related to other types of capability such as surface area capability or material capability. The ontology provides the necessary properties to build connection between different types of capabilities through relationships such as (Cap1: is related to: Cap2), (Cap1: requires: Cap2), or (Cap1:depends on: Cap2). However, it is up to the application-level algorithms to utilize the expressivity of the ontology and estimate capabilities in a more holistic way. Since the intention of this work is to introduce the framework and the ontology, more sophisticated algorithms are not included.

#### *2.5.6.3 Complexity capability:*

Most part quality capabilities in MSDL are directly measurable. The only part quality capability property that is not directly measurable is part complexity. For this purpose, the number of feed axes available on a machine is used to indirectly infer machine's capability with respect to generating complex geometry. An ordinal scale (low, medium, high) is used for part complexity capability measurement. Accordingly, a machine that has less than 3 feed axes has medium to low complexity capability, while a machine with more than 3 axes is considered to have high complexity capability. Obviously, an ordinal scale used in the current implementation is inadequate for complexity measurement in many circumstances but the ontology provides the necessary attributes to support more rigorous measures of part complexity.

#### *2.5.6.4 Material Capability*

Material capability is also inferred based on the submission relationship between different instances of materials. The digital factory contains a list of materials that can be processed by the factory. The built-in reasoner of the CaMDiF framework can identify all super-classes of the explicitly stated material types. The instances of the identified upper level classes are then added to list of materials that can be processed at the factory as inferred materials. The logic behind this approach is that if a vertical mill, for example, can machine a special grade of aluminum then it can also machine more generic grades of aluminum as well.

Material capability, in most real-life scenarios, is evaluated in relation with other capabilities. For example, the grade of material may impact the achievable tolerances and surface finishes on a given machine tool. These dependencies between capabilities can be encoded in the ontology through defining semantic rule which is outside the scope of this paper.

#### *2.5.6.5 Process Capability*

When instances of machines are added to the factory, the manufacturing functions associated with the machines are added to list of available processes in the factory. The functions added directly through the machines are considered to be explicit functions. The CaMDiF reasoner identifies all sub-classes of the explicit functions as the inferred functions. For example, if a machine in a factory has a 'turning' function, then all instances of sub-classes of turning (including boring, facing, grooving, threading) are added to the list of 'inferred' processes (functions) for that factory.

##### *2.5.6.6 Production Capability*

Production capability of a manufacturing facility is related to factors such as production capacity and the variety of the products that can be produced at the facility. Simplistically, the capacity of the factory directly depends of the number of production machines available in the factory. There are some other indirect factors, such as the availability of the preventive maintenance system that can alter the capacity. In this work, only the direct factors are taken into account. Capacity capability class is also measured as an *ordinal measurement datum* with low, medium, high values.

The variety capability, also measured as an ordinal measurement datum, depends on types and variety of manufacturing processes available at the factory. More processes can imply a higher variety of manufacturable parts. Therefore, the system considers both the explicit and the inferred processes when calculating the variety capability of a factory.

### 2.5.7 Modes of Operation

The CamDiF tool can operate both as a desktop application and a web-based application. The proof of concept tool is developed as a desktop application.

### 2.5.8 Software Development Documentation/Design Document

User Manual is provided at the end of this report as an appendix. Other documents related to this project is available on the GitHub page of the project.

### 2.5.9 Users & Use Cases

Depending on the use cases, different users can utilize the developed tools for different purposes. The potential users include supply chain managers at OEMs, manufacturing and design engineers at small manufacturing companies, and analysts at economic development organizations.

- *OEM use case: As a supply chain engineer at an OEM, I need to quickly evaluate and compare a large number of CNC shops in central Texas in order to build a supply chain composed of up to 4 companies for production of a small batch that requires quick turnaround time. I can utilize the framework to evaluate the capabilities of the registered SMMs and pick the ones that closely match with the required capabilities.*
- *SMM use case: As a manufacturing engineer at a small precision machining firm, I can purchase the capability analysis module of the framework and conduct capability gap analysis. I can compare the capabilities of our machine shop with the capabilities of similar shops in the region and use the capability recommendation function to learn about the capabilities and skills that should be acquired by our company in order or remain competitive in the next 5 years.*

## 2.6 Implementation

### 2.6.1 Deliverables 1: MSDL Ontology

The ontology is implemented using Protégé Ontology Development Environment. The OWL file is available on the project's GitHub site.

### 2.6.2 Deliverables 2: CamDiF POC

The implementation resulted in a proof-of-concept (PoC) tool developed in Java. The executable jar file for CaMDiF is also available on the GitHub site. Figure 12 shows the launch screen of the tools


**Figure 12: The launch screen of CamDiF PoC tool**

The Apache Jena framework was utilized to provide the necessary APIs, such as the query and inference APIs, for interacting with the ontology. Jena allows programmers to create and manage various schematic web ontologies using RDF graphs. Jena is employed in the back-end of the CaMDiF software to create, process, and analyze the ontological elements related to manufacturing capability, represented in Web Ontology Language (OWL). The following terms are Jena datatypes most commonly utilized in the CaMDiF software code:

- *OntModel* – a model of an ontology
- *OntClass* – a class resource in an ontology
- *Individual* – an instance of a class resource
- *Property* – an attribute to describe a relation between a subject and an object

The MSDL ontology file is stored in RDF/XML (OWL) format. On startup, the software loads the ontology file into an *OntModel* variable. All necessary classes, individuals, and properties used with Jena are pulled from this OntModel. Exported factories built by users using the software are structured as separate extensions of the core MSDL ontology.

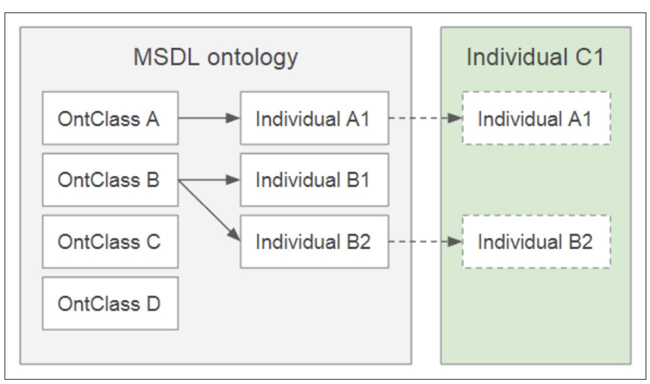



**Figure 13: The structure of the OntClass datatype**

As seen in Figure 13, the MSDL ontology contains various classes (e.g. OntClass A) and various instances of those classes (e.g. Individual A1). User-made digital factories (e.g. Individual C1) are instances of the “factory” class in the ontology. This proof-of-concept implementation uses only six basic categories of individuals when building the digital factory: 3D Printer, Machine Tool, Industry, Material, Software, and Certification. These six classes define the general structure of the user-built factories. Aside from a few other classes that are used for defining meta information about a given company or factory, all Individuals added to a factory fall under these categories (though most Individuals are instances of more narrowed subclasses). For example, the core ontology defines the “EOS-M280” Individual which is an instance of the “DMLS Printer” OntClass. The “DMLS Printer” OntClass itself is a subclass of the “3D Printer” OntClass. A user-made factory may contain the EOS-M280 printer, in which case the factory Individual would be linked to the ontology’s “EOS-M280” Individual through *has factory equipment* property.

#### 2.6.2.1 Build Module GUIs:

The build module provides the necessary functionality for creating user-made factories through adding various resources to the factory. The first step for building a digital factory is to create a company that operates the factory. As shown in Figure 14, the user has also the option of editing the existing factories that are already saved in the system or importing factories through browsing and loading their factory.owl files. Figure 15 shows the user interface designed for the purpose of creating a new factory. The meta data, such as name, address, url are regarded as the instances of the *textual entity* class (a sub-class of *generically dependent entity* in BFO) that stand in *aboutness* relationship with the company. The URL of the company is later used by the *analyze* module to crawl the website of the company and extract capability-related entities.


**Figure 14: Possible options under the Build module**

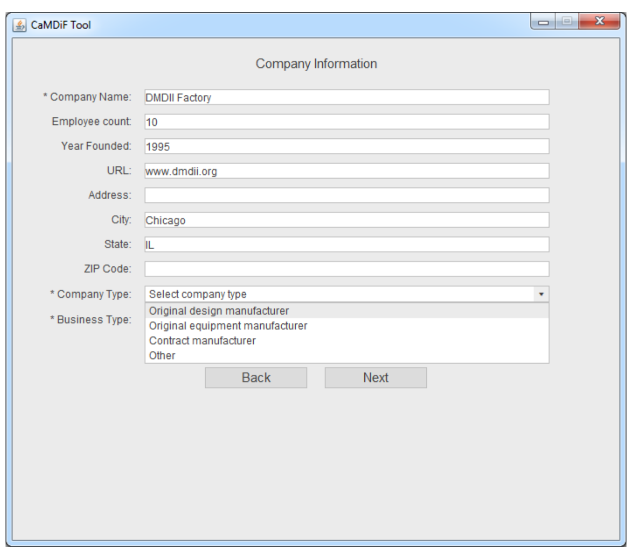



**Figure 15: the interface for entering company information**

The next step is to add different types of equipment to the factory. In this PoC tool, only CNC machines and 3D printers are used as factory equipment. The ontology contains libraries of CNC machines and 3D printers as ontological instances. As shown in Figure 16, the user can browse the class structure of machine tools and then view the available machine instances for the selected machine type. The properties of the selected machine can also be inspected by pressing the "Info" button. Machine properties are maintained as CSV (Comma Separated Values) file and then imported to the ontology using the *celffie plugin* of Protégé. The specifications of the machines in the library are collected form the machine catalogs published by machine vendors. The selected machines then can be added to the factory one by one. It should be noted that no new instance will be created in this process and the user only selects the available machine instances. Figure 17 shows the vertical milling machine and 3D printer instances added to the user-defined factory.

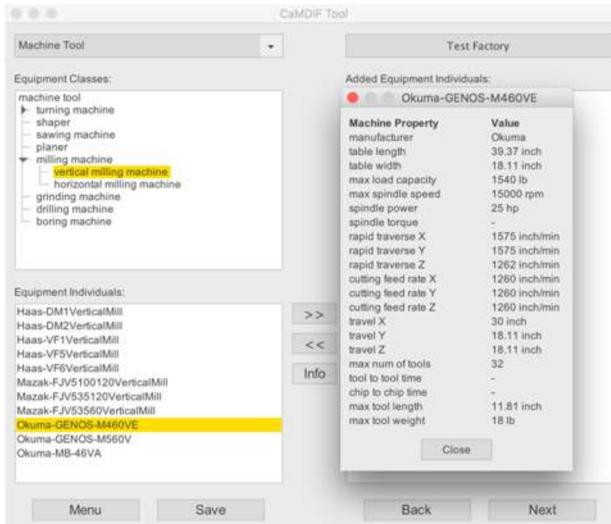



**Figure 16: The interface for adding Machine Tools to the factory**

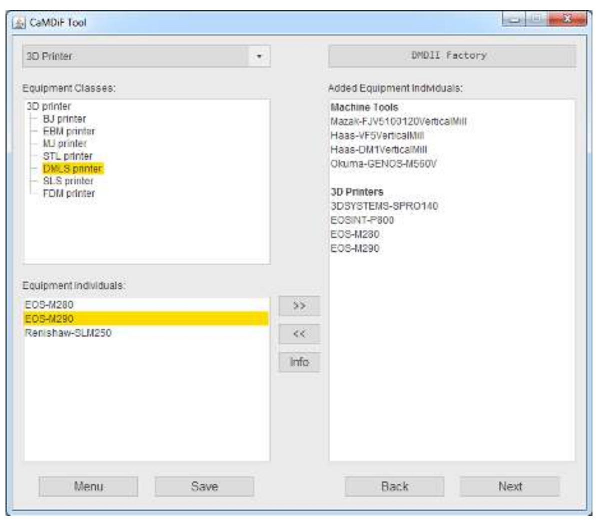



**Figure 17: The CNC machines and 3D printers added to the user-defined factory**

The user can also add other entities such as material, industry, and software application to the factory as shown in Figure 18. These entities are also used when drawing conclusions about the capabilities of the factory. The user can add the skills available in the factory to the capability model of the factory (Figure 19).

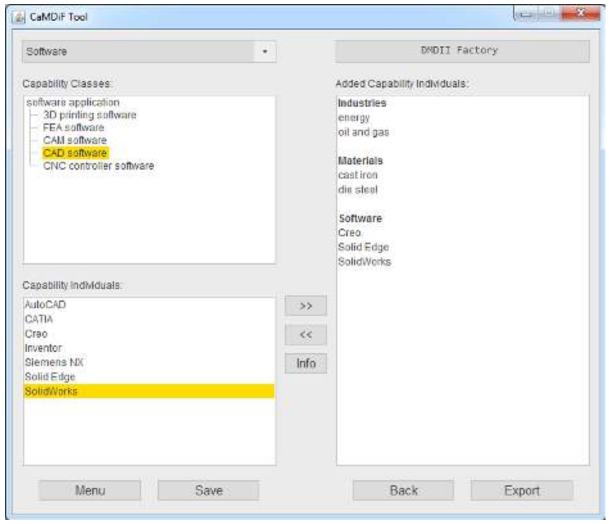



**Figure 18: The interface for adding material, industry, and software capability to the factory.**

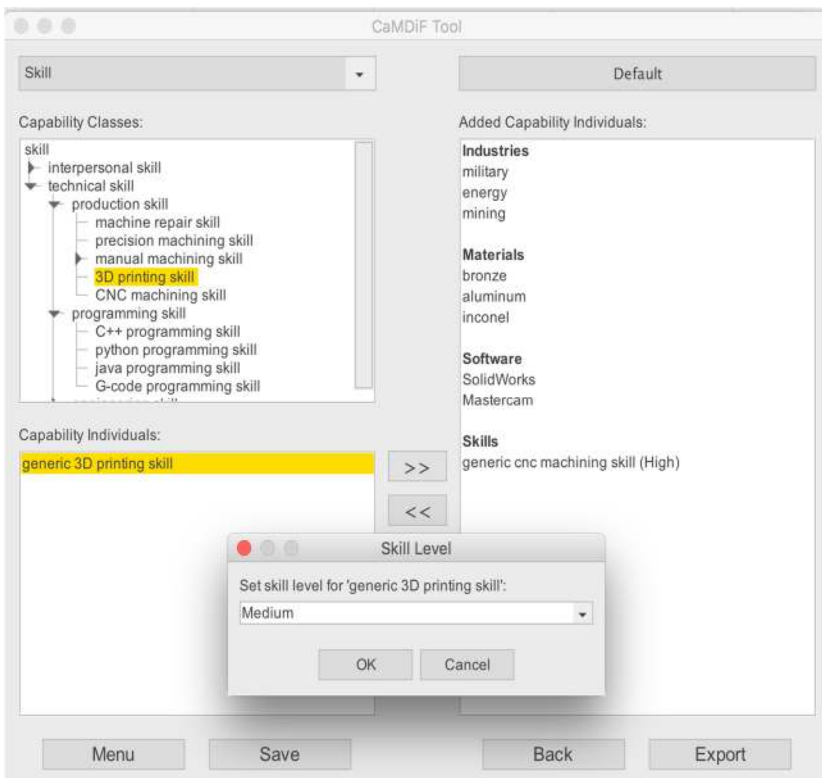


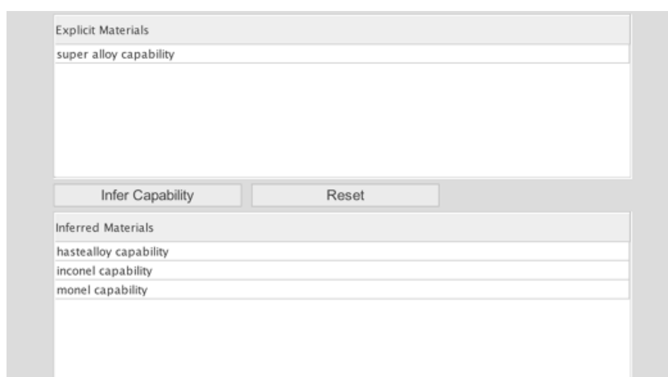

**Figure 19: The interface for adding skills to the factory**

Once the necessary entities are added to the factory, the resulting owl file can be saved locally or exported as an XML/RDF file. Since this file uses standard syntax and semantics, it can be consumed by third-party applications for different purposes.

#### *2.6.2.2 Analyze Module GUIs:*

The main function of this module is to interpret the manufacturing capabilities of the user-made factories using Jena's reasoning functionality. The analysis begins with selecting an existing factory or importing an external factory already represented in MSDL. The first tab, depicted in Figure 20(a), provides information about the factory's part quality capabilities such as the achievable tolerances and surface finishes. If part quality capabilities of the factory equipment are not available explicitly, the Jena reasoner looks for the most immediate super-classes of the factory equipment and uses their capability information to infer the part quality capabilities of the factory. The second tab shows process capabilities. The inferred process capabilities are sub-classes of the processes explicitly included in the factory. For example, if the explicit process is vertical milling, then face milling and end milling operations are returned as the inferred processes.


**Figure 20: (a) part quality capability tab (b) process capability tab**




a


b


c

**Figure 21: (a) material capability tab (b) production capability tab (c) extracted capability tab**

Figure 21 (a) and (b) show the material and the production capability tab respectively. In this example, super alloy capability is provided by the *factory.owl* file as the explicit capability but the reasoner has concluded that Hastelloy, Inconel, and Monel can also be added to the list of materials that the factory can process. Also, although the explicit variety capability is low, the reasoner infers medium capability level for variety since the inferred manufacturing processes are more diverse that the explicit ones. As mentioned before, variety of parts producible at a factory directly depends on the variety of available processes.

The Extracted Capability tab shown in Figure 21 (c) lists the capability features that are extracted from the website of the company which operates the factory. The capability features (concepts) come from the SKOS thesaurus of manufacturing capabilities. Each column heading is a *top concept* in the thesaurus and the concepts listed in each column are the narrower (more specialized) form of their top concept. This tab provides more information beyond what the factory.owl file can provide about the manufacturing capabilities of companies. Once the analysis process is complete, the user can export the capability file that includes both explicit and inferred capabilities. The extracted capability entities from the company website are also included in the exported capability file.

#### 2.6.2.3 Match Module GUIs:

The match module provides the necessary functionality for building supply chains for a given work order (WO). The first step in this module is to create a work order through describing the part attributes and the expected quantities. Then the desirable capability features (provided the capability thesaurus) will be added to the work order. These features will be used in the next steps for feature-based optimization of feasible supply chain. The last step is building the supply chain based on the available digital factories in the factory repository of CaMDiF. The uses can set the size of supply chain (currently ranging from 1 to 4 factories). The generated supply chains can also be exported as an OWL file with XML/RDF syntax.

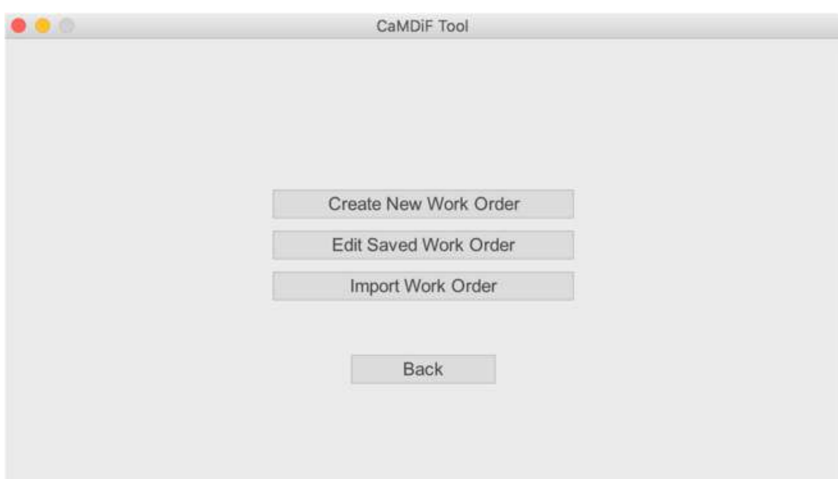



**Figure 22: The launch page of Match module**

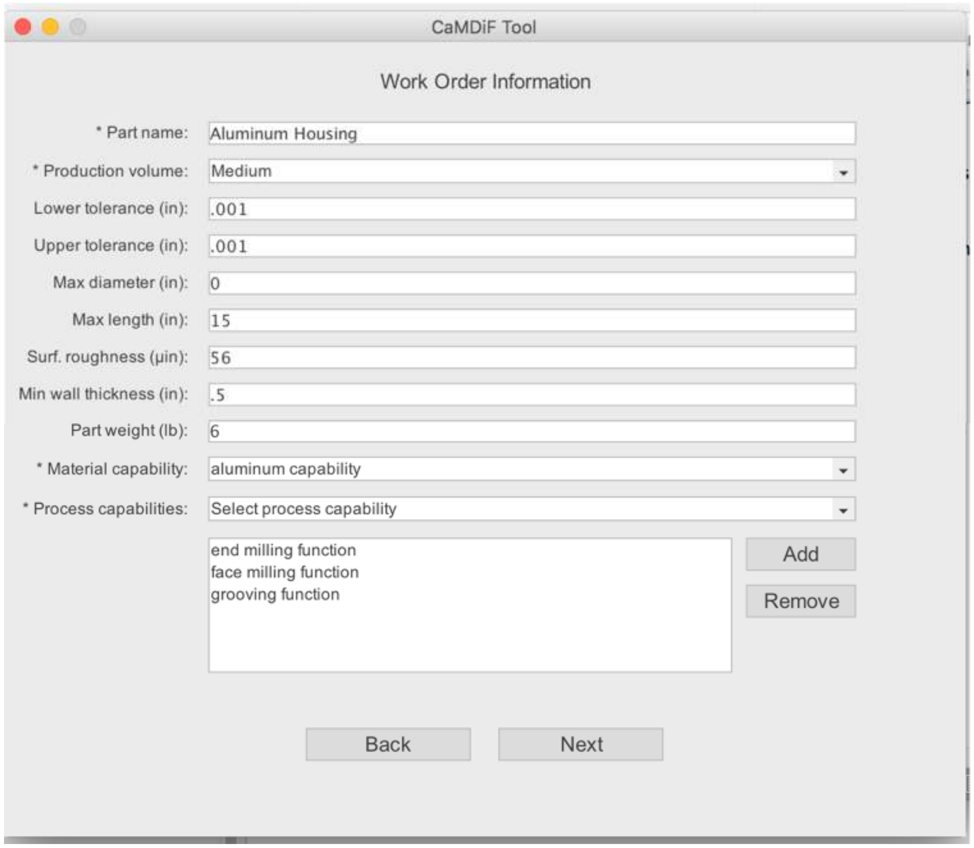



**Figure 23: The UI for entering the part information related to a WO**

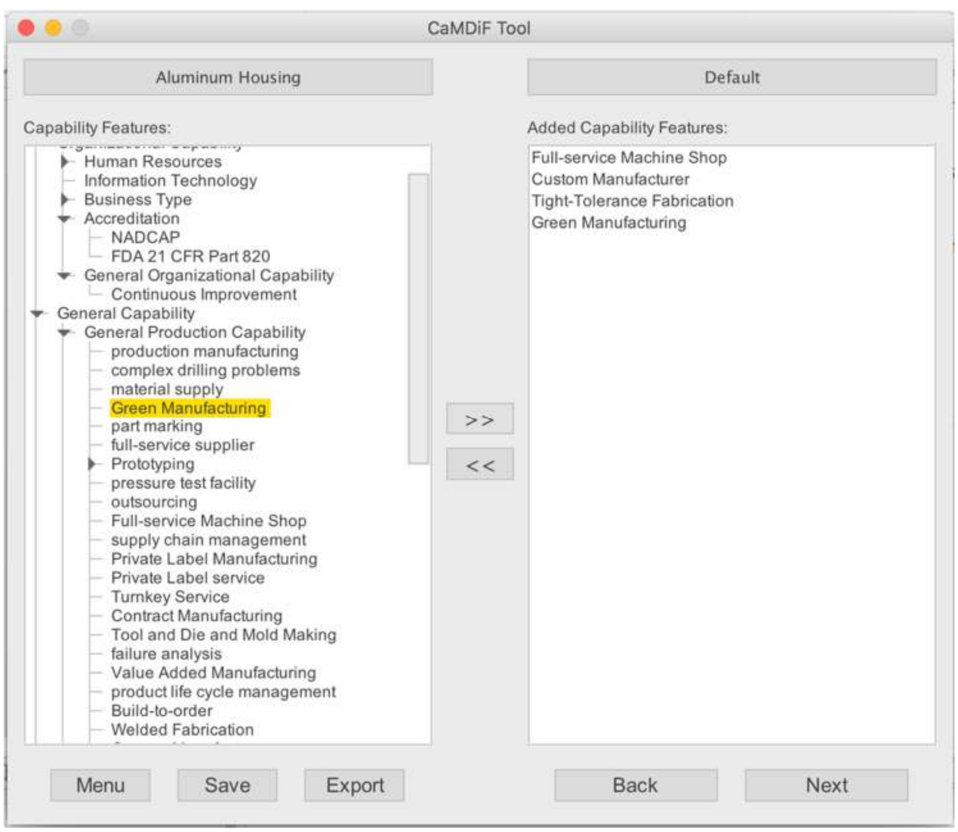



**Figure 24: The UI for adding capability terms (desired capability feature) to the WO**

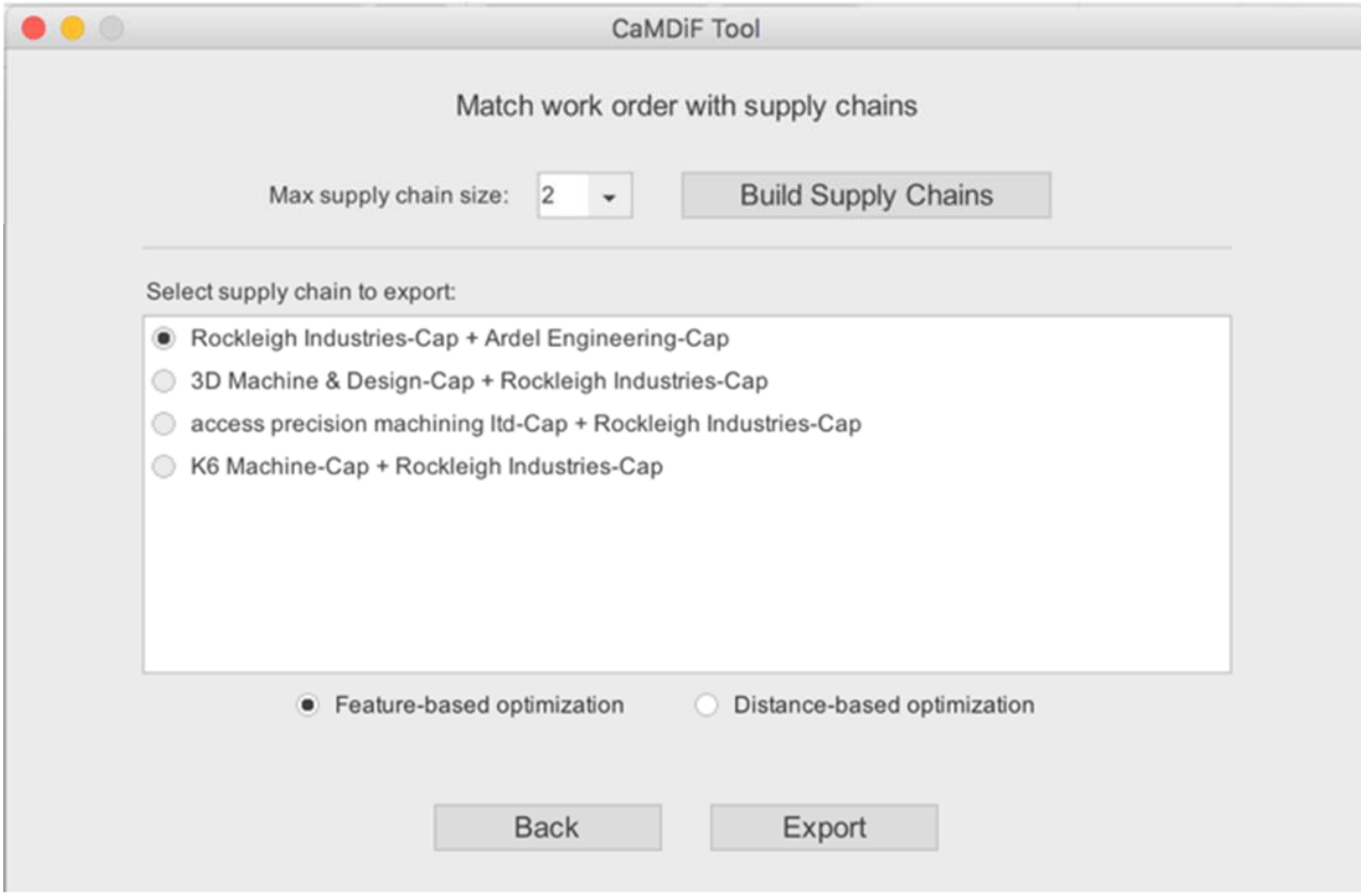



**Figure 25: The UI for building supply chains for the given work order**

## 2.7 Tech Transition Plan & Commercialization

### 2.7.1 Identify Future Plans

In terms of enhancing the functions and features of the software (in order to make it a commercially viable solution), the following actions will be taken:

- The POC tool will be reconfigured and reprogrammed as a web-based tool
- The library of CNC and 3D printing machines will be extended to cover all of the machines from all major vendors
- Include other manufacturing processes such as casting and metal stamping.
- Extend the ontology to include new processes
- Extend the capability reasoning algorithms:
    - Account for relational capabilities. For example, what are the tolerance and surface finish capabilities of this machine in relationship with these particular materials.

### 2.7.2 Tech Transition

Small to Medium Size Manufacturers were interviewed to solicit input and determine their level of interest in implementing CAMDif. As suspected, the responses were indifferent. For SMMs, the technology needs to be quick, easy, and affordable. If you can't show them how you are going to immediately increase their revenue or profits, they simply don't want to invest any time at all in evaluating or learning a new type of software. You need to be able to quickly tell them how your solution will immediately impact their bottom line. Since the outcome of our project was a 'proof of concept', it is not ready to be transitioned into the marketplace. There are several factors that need to be improved before we can test market acceptance.

1) **Build Module**
   - *Large, Accurate Resource Element Libraries* - Users must quickly find their machine, software, 3D printer, etc. and the resource must exactly match what they have. A tool could possibly be created to automatically determine the Resource Elements of each company.
   - *Search Functionality* – As the Resource Element Library grows, the need for search and filtering will be needed.
   - *Interfacing Capability* – the ability to interface with other inventory or business systems (ERP, MES, Hoovers, etc.) would be useful to speed up the 'building' of the digital factory.
   - *More Detailed Categories* – Software should be further categorized into the different modules of each software. For instance, "MasterCAM" is too broad. There are over several different MasterCAM modules that contain different functionality for 3, 4, or 5 axis machine center programming. The capability of the factory could be limited due to the lack of software functionality.
   - *(People) Skill Categories* – The skill categories should be based on the type of Resource Elements that are contained in the library. For instance, if the factory has "MasterCAM" software, the skill list should have categories based on 3,4 or 5 axis functionality and how experienced the user is with the software. Similarly, the skill list should be based on the type of CNC machines that are being used.
   - *(People) Skill Levels* - The skill levels (very low, low, medium, high, very high) are too subjective. This could be based on the type of certifications (ie NIMS) or training classes completed. Test could also be provided to gauge a person's skill level.
   - *Additional Categories* – Additional categories are needed to determine capability more accurately. A Product Experience category could be created to determine familiarity with product types (weapons, tractors, pistons, golf clubheads, etc.). Cutting Tools, Cutting Tool Holders, Workholding, and Coolant categories will better determine the CNC machining capabilities.
   - *Standards Based* – The resource library elements need to be based on industry standards. There are current standards for materials, people, and products that could be utilized. A working relationship with MTConnect, ISO, ANSI, or other organizations should be formed to ensure CamDif will evolve with standards that are already in place.

2) **Analyze Module**
   - *Part Quality Capability* – This could become more accurate with physical machine capability testing (ball bar, laser interferometer, etc.). If a factory has a particular machine, it does not necessarily mean that it can manufacture parts to a particular tolerance. The machine could contain a bad bearing, excessive backlash, or other reasons that it cannot adequality manufacture as intended. If physical testing on the machine was performed, we could substantiate the Part Quality Capability.
   - *Process Capability* – for CNC machining centers, this could include 3,4 or 5 axis and mill-turn capabilities. The Process Capabilities could also be matched with toolpath types (drill, mill, threading, turning, reaming, etc.)
   - *Material Capability* – this should match industry standards.
   - *Production Capability* – this could be a summation of the number of CNC machines that a factory has multiplied by the hours of machine time available. The variety should be based on the type and size of material that can be manufactured and the process capability (# axis, toolpath type, etc.)
   - *Extracted Capability* – The extracted capability is good to get a general idea of what a company does. Although, this functionality could result in poor accuracy of a company's capabilities. For instance, a company could have the word 'milling' on their website but only have manual mills.
   - *Save and Export* – in the web-based version, this information should automatically be stored in the cloud.
   - *Compare Factories* – Material, capacity, and additional categories could be added. Functionality could be made to add multiple factories to determine regional capability and capacity. The tool would then be valuable to economic development organizations.

3) **Match Module**

- *Work Order* – the ability to upload CAD files will significantly enhance usability. CAD files based on MBD (Model Based Definition) standards could automatically determine the tolerances, weight, surface roughness and other critical information that is needed. Feature recognition software could also be used to determine the capability that is needed for each part.

4) **All Modules**

- *Software-as-a-Service (SAAS)* – CamDif needs to be available in any web browser. The user will not want to download and install software.
- *Accuracy* - This is the most important criteria for technology transition. CamDif must build, analyze, and match very accurate capability models or the users will abandon the software quickly.
- *Affordability* - less than $20 a month or free.
- *Usability* – The user interface needs to be improved with drag-and-drop functionality and provide data visualization of the digital factories and supply networks.

While our original intentions were to develop CamDif for supply chains, there was interest in using the tool for internal use. Rock Island Arsenal currently has over 150 CNC Machines. Their people that develop process plans are often not familiar with the capabilities of each CNC machine. Therefore, they often designate the same machine, machines that have the most capability, or incapable machines on their process plans. This leads to capacity issues on their most capable machines and the need to reassign machines when a part cannot be made on the originally assigned machine. With CamDif, they could automatically determine exactly which machines would have the ability to make a part. The software would also show them alternate machine choices when capacity becomes an issue.

Overall, the users liked the potential benefits of CamDiF. After additional improvements are made and quality testing performed, it will be a great tool to determine internal capability or to develop new supply chains.

### 2.7.3 Commercialization plans

A team of developers hosted at Texas State's STAR Park will work on the project during the commercialization phase (not part of the current Enterprise Project). The objective of the first phase of the commercialization plan is to developed a web-based version of CaMDiF. More functions and features will be added as well to the platform in this phase. The necessary protocols for enabling third-part apps to use CaMDiF capability repositories will be developed. The Science, Technology, and Advanced Research (STAR) Park is a 58-acre site that hosts STAR One, Texas State's first building dedicated to the university's research and commercialization efforts. The 36,000 square foot facility serves as a technology incubator for start-up and early-stage businesses.

The Innovation Machine will continue to work with the technology to prepare it for commercialization. Commercialization efforts will focus on perfecting the solution with one manufacturing process (CNC Machining) and an internal system. Once perfected, it can then be scaled to additional processes and larger networks. Since a need has been identified for CNC Machining at Rock Island Arsenal, our efforts will be focused on a pilot implementation project there. We will focus on the accuracy and usability of the CNC Machining Capability functionality. Once the aforementioned improvements have been made, we will work with the Quad Cities Manufacturing Innovation Hub to pilot a project for their Regional Capabilities Catalog.

We would like additional large OEM manufacturers to participate in our follow on pilot projects. The OEMs that participate will receive the benefits of CamDif and also obtain the confidence that the tool is accurate. They can then utilize the tool with their internal supply chains.

### 2.7.4 Market Assessment

Three target markets have been identified for CamDif.

1. **Suppliers** of component parts (machined, forged, cast, etc) could use the technology internally to determine the capability of their machines and externally for their customers (OEMs). The technology would be useful to any supplier or OEM of component parts. This represents thousands of companies. For instance, there are 18,235

machines shops in the United States. This one process only represents a fraction of the suppliers that CamDif could serve.

2. **Original Equipment Manufacturers (OEM)** that purchase component parts could use the technology to develop their supply chains. In the United States, there are 75,188 companies in "Fabricated Metal Product Manufacturing". This only represents a fraction of the market.
3. **Economic Development Organizations** could use the software to determine the strength and weaknesses of their regional capability. The regional capability mapping would be a unique selling advantage to help recruit companies and new revenue. The capability mapping activity would also assist each company in documenting their Resource Elements (RE)s, the first step in becoming a 'Digital Factory'.

### 2.7.5 Identified Barriers to Adoption

Small to Medium Size Manufacturers (SMMs) will not adopt the software unless there is clear evidence of how it will increase their revenue or profitability. CamDif will be demand driven – unless you have several large OEM Manufacturers using the tool to find suppliers, SMMs will not participate.

The building of the digital factories must be incredibly simple. SMMs will spend very little time documenting their Resource Elements. There needs to be a focus on making the Digital Factory "Building" quick, easy, and affordable.

Economic Development Organizations would need nearly all companies to participate in their capability mapping program. Without funding or a clear SMM incentive, they will lack the adoption rates needed to rate an entire regions capability. Adoption by 3rd party application developers is also necessary to ensure widespread adoption by SMMs.

## 2.8 Workforce Development

Since our product was focused on only a 'proof of concept', limited time was spent on Workforce Development efforts. A CamDif User Manual was developed to assist first time users of the software. Once the web-based version is

complete and the CamDif product is more mature, we will develop online training resources (help menus, videos, support contact, etc.). Workforce Development seminars could take place at OEM Supplier conferences or Economic Development Offices.

## 2.9 Project Success Criteria

Table 3 summarize the success criteria for the project.

- Functionality: To test if the Build Function of the developed tool works effectively, more than 20 digital factories corresponding to real suppliers were generated using the POC tool. Human experts compared the configuration of generated factories with suppliers' resources and it was determined that the digital factories replicate their corresponding physical factories reasonably well. Also it was determined that the ontology is capable of modeling almost all of the critical manufacturing capabilities in the context of project's use cases.
- Precision: The *Match* function of the proof of concept was tested through evaluating the quality of the returned suppliers based on expert's judgement. It was observed that most of the matched suppliers (85%+) had the necessary qualifications to fulfill the given work orders. The precision was tested based on a sample of 5 work orders and 30 digital factories.
- Scalability: Since the POC tool is using a different technology compared to the web-based version, the results of scalability test on the POC will not be applicable to the final solution.
- Extensibility: Based on experimental modeling of different manufacturing processes, it was determined that the ontology has no limitation with regard to accommodating different types of manufacturing processes such as casting and forming.

**Table 3: project success criteria and their related metrics**

| | Success Criteria | Metric | Result |
|---|---|---|---|
| **Functionality** | Does the ontology properly capture all types of manufacturing capability? | % of the capability types that can be modeled | 100% |
| | Does the digital factory correctly reflect the capabilities of the physical factory? | modeling accuracy based on the expert's judgment (low-med-high scale) | High |
| **Precision & Accuracy** | Does the matchmaking engine correctly connect supply and demand entities? | Precision and recall measures for search query. 75% + expected | 85-90% |
| | Does the generated supply chain have the required capabilities? How accurate capability measurement is? | Quality of the generated supply chains as judged by experts (low-med-high scale) | High |
| **Scalability** | Is the provided solution scalable enough to accommodate large-scale factories and supply chains? | Size of the factory & supply chain that can be accommodated | Not evaluated |
| **Extensibility** | Can the developed models be extended to all types of manufacturing processes? | % of manufacturing processes that can be modeled | 100% |

## 2.10 Conclusion

This report presented CaMDiF as a software system for building supply chains automatically supported by semantic ontologies, standards, and data analytics tools. CaMDiF can create an ecosystem of manufacturing capabilities that are formally described and can be shared, discovered, evaluated, and integrated autonomously. The capability model of the factory can be derived through ontological reasoning based on the available resources in the factory. The proposed model of Digital Factory meets the needs of the factories of the future, as they will be reconfigurable, adaptive, and evolving. One of the advantages of Digital Factory is real-time and accurate representation of the technological capabilities of manufacturing companies.

Through exploring and querying the Digital Factories and their associated capability models, companies can develop a deeper and more precise understanding of the technological capabilities of prospective suppliers, thus making more informed decisions when building supply chain partnerships. By creating their digital twins, small to medium-sized manufacturers can significantly improve their visibility in the virtual space. This, in turn, will increase improve their revenue and profitability. OEMs can also use Digital Factories internally, for assessing the capabilities of their various plants, or externally for evaluating the capabilities of prospective suppliers.

Since Digital Factories are described semantically using MSDL ontology, they are amenable to automated search and reasoning. Also, conformity of the ontology with BFO enhances interoperability and semantic stability. The generated capability model can be used by third-party applications for different use cases such as sourcing or manufacturability analysis. Schema.org experience shows that a key driver for high level of adoption of a vocabulary is extensive support from third-party applications who commit to the vocabulary. The capability ontology is dynamic entity and will be extended to capture different aspects of manufacturing and production capability. In future, digital capabilities, and human capabilities (skills) will be added to capability model of MSDL. Also, the capability analysis module will be extended to provide capability recommendation services which suggests new and supplementary capabilities to suppliers depending on the available work orders in the demand pool of the market. The future versions of CaMDif will be able to address capability dependency as well. For example, surface area producible and surface finish producible are two capabilities that may be inversely related. These types of relationships, although already captured by the ontology, should be incorporated in capability reasoning and analysis procedures. Another extension is providing the user to create user-defined and configured machines in CaMDiF environment. Currently, the tool only supports standard machines as configured by their vendors. Also, there is a need for introducing more efficient and scalable mechanisms that can be used for directly importing machine specification, provided by the vendor, into the ontology.

## 2.11 Lessons Learned

### 2.11.1 Problems Encountered

One of the issues we encountered during the validation phase was that different industry users had different perception of manufacturing capabilities. This resulted in a slow down at the early stages of the experimentation. After semantic onboarding, communication was significantly streamlined.

### 2.11.2 Proposal Claim Deviations

<u>Extra contribution beyond the original SOW:</u> To enhance the intelligence of the platform, a thesaurus of manufacturing capability terms was added to CamDiF. This thesaurus was not included in the original proposal and was added as an extra module beyond what was promised in the proposal.

When building sophisticated knowledge structures for the purpose of rigorous reasoning becomes necessary, development of formal axiomatic ontologies is unavoidable. However, in many occasions, terminological services are sufficient to enable some light-weight inferences. A thesaurus can serve as a knowledge graph that represents the relationships between various entities in a knowledge-base. For this purpose, a thesaurus of industrial capability terms was developed to be added to the knowledge layer of the proposed framework. Although the thesaurus terms can be mapped to ontological classes, in this work, thesaurus and ontology are regarded as two separate information entities that work independently. The particular application of the thesaurus in this work is to tag the extracted text from the website of manufacturing companies with capability-related terms and phrases.

The Manufacturing Capability (MC) Thesaurus captures the terms that directly or indirectly point to an aspect of manufacturing and/or production capability. Contract manufacturers may use terms and phrases such as precision machining, tool and die making, or build-to-order manufacturing to explicitly describe their technical capabilities, expertise, and services. Also, they may provide examples of parts they have produced in the past or industries and customers they have served in the past to advertise their capabilities indirectly. In the presence of a comprehensive thesaurus of manufacturing capability terms, it is possible to readily translate each website into a vector model that is more amenable to quantitative analysis. The thesaurus uses Simple Knowledge Organization System (SKOS) formalism which is a W3C standard for representing formal thesauri. SKOS provides a structured framework for creating different types of controlled vocabulary such as thesauri, concept schemes, and taxonomies. Figure 26 shows the concept diagram for *Swiss Machining* process based on the SKOS semantics.

Each concept in SKOS has exactly one preferred label (*skos:prefLabel*) and can have multiple alternative labels (*skos:altLabel*). *Screw Machining* is the alternative label for *Swiss Machining* as it is used frequently for referring to the same concept. The broader concept *(skos:broader)* of the *Swiss Machining* is *Machining*. *Swiss Turning* and *CNC Swiss Machining* are the narrower *(skos:narrower)* concepts; meaning that they are more specialized forms of Swiss Machining. The broader and narrower relationships are included in SKOS semantics to encode hierarchical relationship between concepts. The concepts that are related to *Swiss Machining* through *skos: related* include *automatic screw machine*, *small part*, and *swiss type turning*.

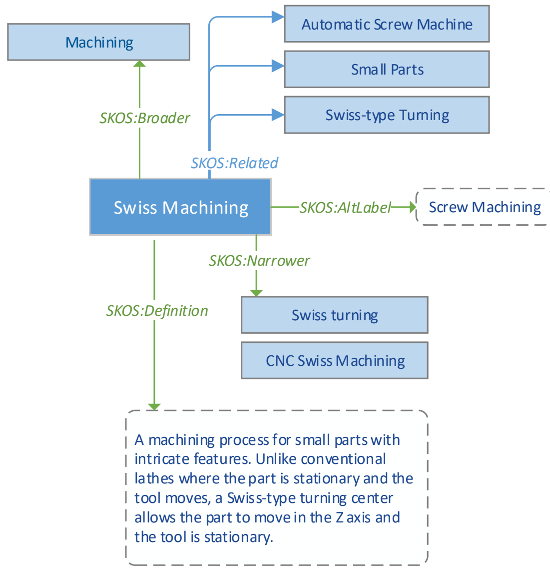



**Figure 26. The SKOS concept diagram for Swiss Machining process**

The reason *small part* is a *related* concept to Swiss Machining concept is that Swiss Machining is typically used for machining small parts with intricate features. In addition, each SKOS concept can have a definition provided in plain English or any other natural language. One major advantage of the SKOS thesauri is that they can be extended by community crowds and shared as linked data due to their open and standard syntax and semantics. It should be noted that there is no connection between the MC thesaurus and MSDL ontology and, in CaMDiF framework, these two information artifacts act independently. However, it is possible to linked them through semantic mapping.

Other Deviations: We were initially planning to use Microsoft Technology infrastructure for app development but due to the closed-source nature of Microsoft Technology and lack of support for Semantic Web technology, we used a different platform (Apache Jena) for application development. Also, we were initially planning to host the developed apps on Digital Manufacturing Commons (DMC) but due to limited uptake of DMC, this plan was not materialized.

### 2.11.3 Risks

# 3 Appendices:

## 3.1 Manufacturing Capability Self-Assessment Report

| | |
|---|---|
| Company Name : | **ITAMCO** |
| Department/Unit Name: | |

*The objective of this assessment is to document the company's perception of its manufacturing capabilities. The capabilities captured through this assessment are similar to the capability information publicly available on the company's website. You can either do a single self-assessment on the entire company as a whole or prepare separate assessments for different departments/units within your company. Feel free to include additional categories if you believe the provided categories do not cover your core capabilities.*

**Capability Range:**

Specify the values for different capability parameters listed below. Feel free to add more capability parameters as you find appropriate for your facility.

*Quantitative Parameters*

| Parameter | Value |
|---|---|
| Precision (inch) | .0001(+-) |
| Max part length | 100" |
| Min part length | 1" |
| Max part dia | 120" |
| Min part dia | 1" |
| Minimum wall thickness | .100" |
| Surface finish (micro-inch) | 1 RMS (REM Superfinisher) |
| Other parameters | |
| Other parameters | |
| Other parameters | |

*Qualitative Parameters*

| Parameter | Value (low-medium-high) |
|---|---|
| Part Complexity | high |
| Part Variety | high |
| Production Range | low |
| Other parameters | |
| Other parameters | |

### Process Capability:

List the manufacturing processes your facility can provide. Please be as specific as possible. Include all specialized processes.

| |
|---|
| e.g.: Milling, Boring, Turning, OD/ID Grinding , Gear Cutting, Gear Grinding, Gear Hobbing, Gear Shaping, Deep Hole Drilling, plating, coating, welding, bore welding, EDM, DMLS, shot peening, coal blasting, sawing, nital etch, MPI, REM Superfinishing |

### Material Capability:

List the materials that can be processed at your facility. Please be as specific as possible. Include all specialized materials.

| |
|---|
| 9310, 8620, 4340, 4140, Carbon Steels, tool steel, Die steel, Aluminum, exotic alloys |

### Part Capability:

List the types of parts that can be manufactured at your company. Provide some representative examples.

| |
|---|
| gears, valves, housings, rolled rings, manifolds, carriers, etc. |

### Capability Class:

Describe the specific capabilities your facility can provide. Some examples are provided below. Think of them as the advertising phrases/labels you use on your company website to describe the capabilities of your company. You may include non-production capabilities as well such as engineering design.

| Capability Class |
|---|
| Heavy and bulky part machining |
| Long and Large part diameter machining |
| Low tooling cost |
| Mass customization |
| Turnkey production |
| Custom fixture design capability |
| Comprehensive product testing |
| Additive Manufacturing |
| Large Assembly |

### Capability Enablers:

List the resources that have enabled the capabilities you identified in this document. Your capabilities can be traced back to your production equipment, human recourses, software applications, industry certification, etc. Some examples are provided below

| Capability Enabler |
|---|
| 5-axis CNC machines |
| CMM |

| |
|---|
| CAD applications (Autodesk) |
| Skilled machinists |
| ISO 9001:2008 certification |
| Cleanroom |
| DMLS printers |
| |
| |

## 3.2 Capability Dependency Form

# Capability Dependency

Company Name : ______________________________

Please identify the important manufacturing and engineering capabilities of your company. For each capability, list other capabilities, skills, and resources that support/enable the identified capability (one table per capability).

| **Capability:** large custom gear machining capability (Example) | | |
|---|---|---|
| Requires capabilities, skills, and resources<br>⇩ ⇩ ⇩ | | |
| **Capability** | **Human Skill** | **Resources**<br>**(Machine, tool, software, transportation equipment, test equipment, etc)** |
| Heavy part inspection capability<br>Heavy part transportation capability | Machining skill<br>Gear cutting skill<br>CMM operating skill | Vertical machining center<br>Niles gear-profile grinder<br>CMM<br>Lift truck<br>Hoist |

| **Capability:** insert the capability description here | | |
|---|---|---|
| Requires capabilities, skills, and resources<br>⇩ ⇩ ⇩ | | |
| **Capability** | **Human Skill** | **Resources**<br>**(Machine, tool, software, transportation equipment, test equipment, etc)** |
| | | |

April 2018

# User Manual

CaMDiF

Capability Modeling for Digital Factory

Farhad Ameri

*Questions: ameri@txstate.edu*

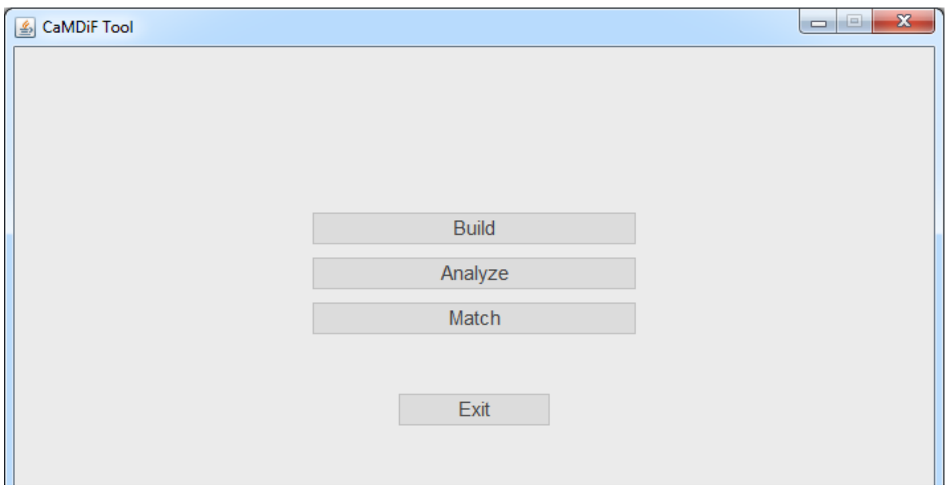
## 3.3 User Manual

CaMDiF is a software framework that enables manufacturing companies to describe and share their manufacturing capabilities using a standard format. This document provided detailed instructions on how to use the CaMDiF tool for creating instances of digital factory and analyzing the capabilities of the created factories.

### 3.3.1 running the jar file

Download the executable jar file and double click on it to launch the program. The provided executable jar file runs on both Windows and Mac OS x platforms. The launch screen of the tool provides the user with three options:

- **Build**: building a new digital factory or editing the existing factories
- **Analyze:** analyzing the manufacturing capabilities of digital factories
- **Match**: matching a work order with one or more digital factories that have the necessary capabilities to fulfill the order

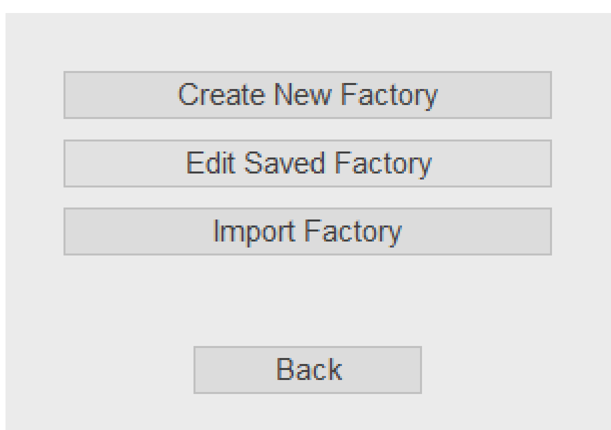



# BUILD

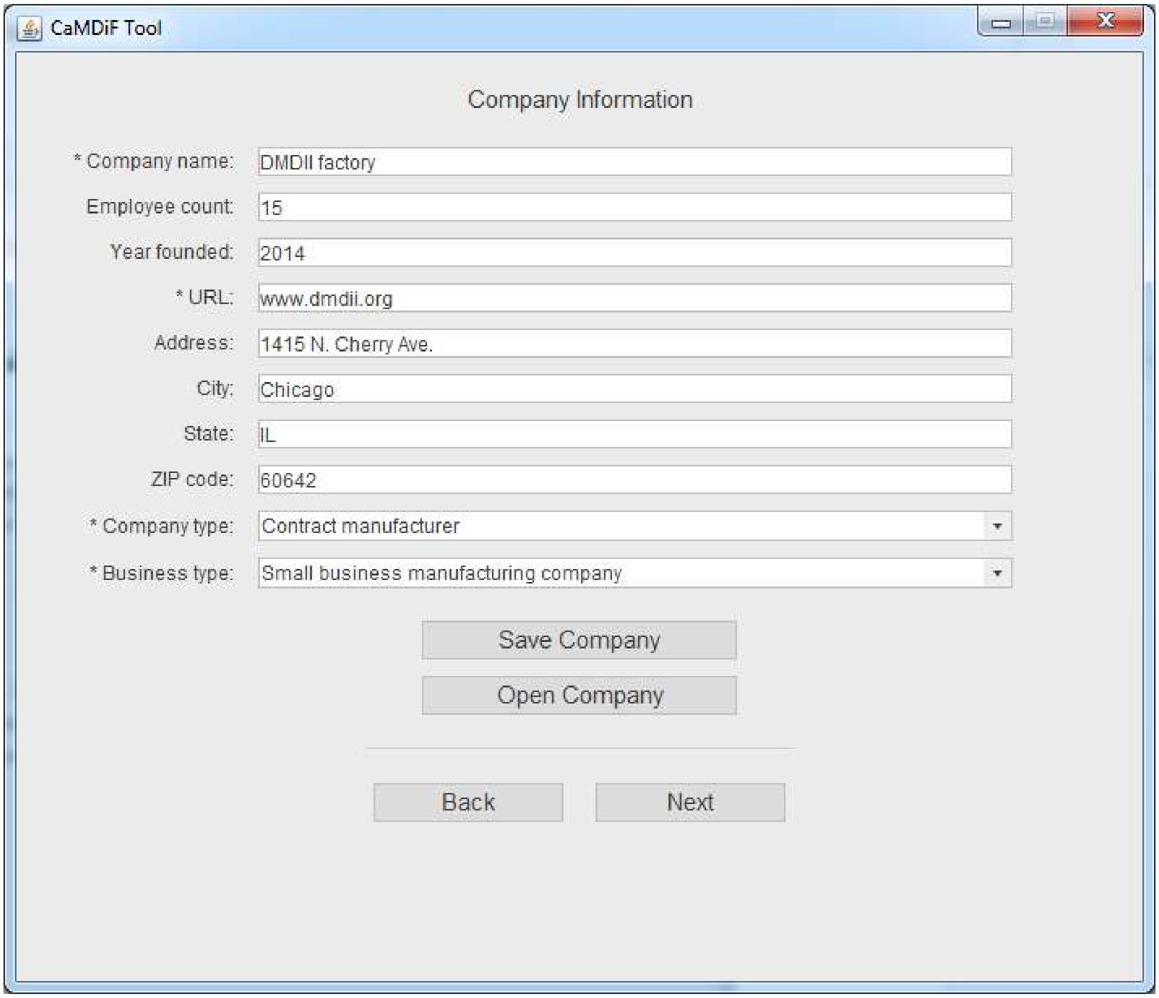
The build module provides the necessary functionality for creating user-made factories through adding various resources to the factory.

### 3.3.2 create new factory

The first step for building a digital factory is to create a company that operates the factory.

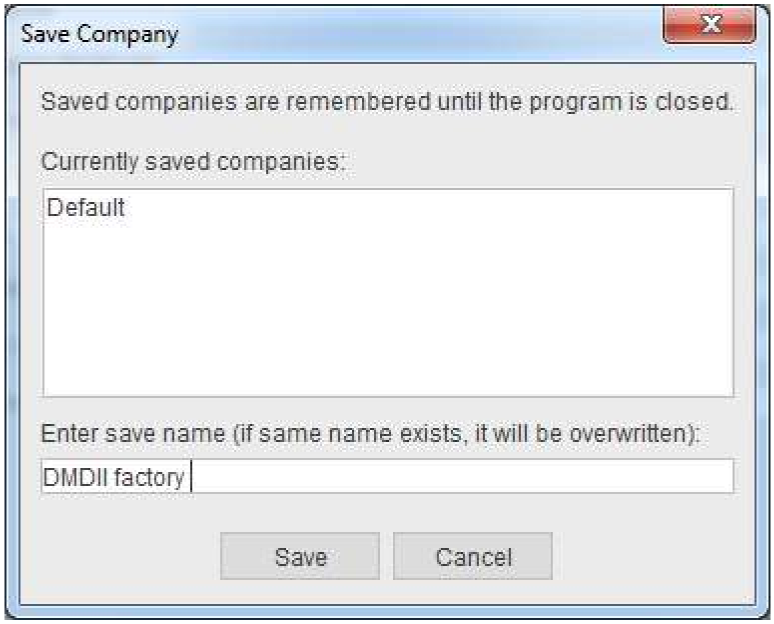


1. **Create New Factory:** Enter the requested information about the company that operates the factory. The fields with asterisk are required fields. Alternatively, you can edit a saved factory or import a factory (factory file with .owl extension).




2. **Save Company:** Save the company after the necessary information is provided. The name of the company will be used as the name of the company file. Note that once the program is closed, the saved factories will be deleted.

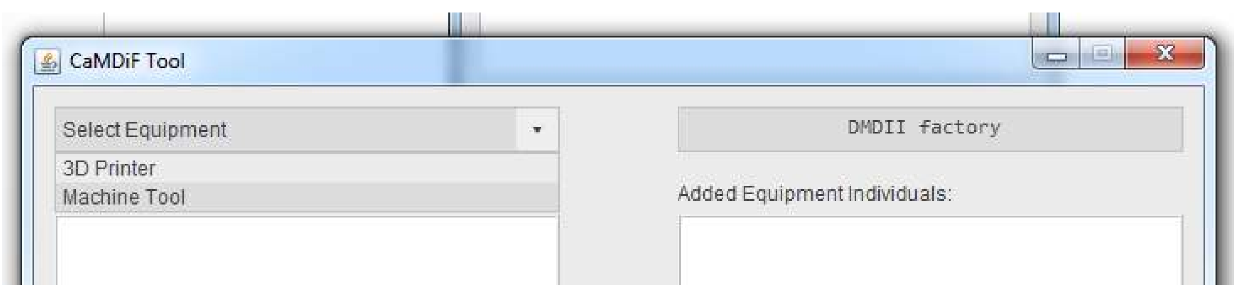
   **Open company:** Use this button to open a saved company within the current session.



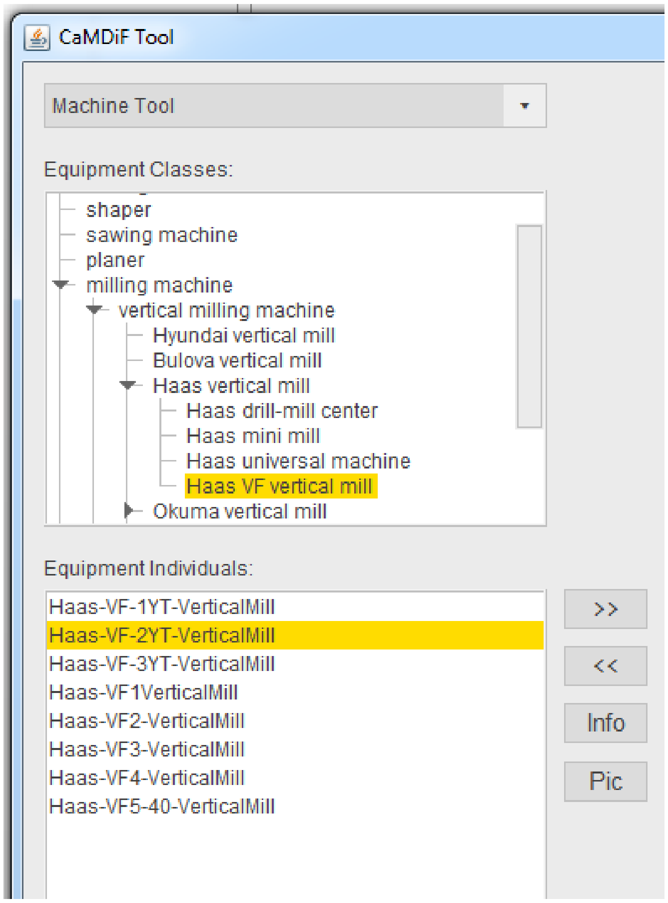

3. Press “Next” to go to the next step.

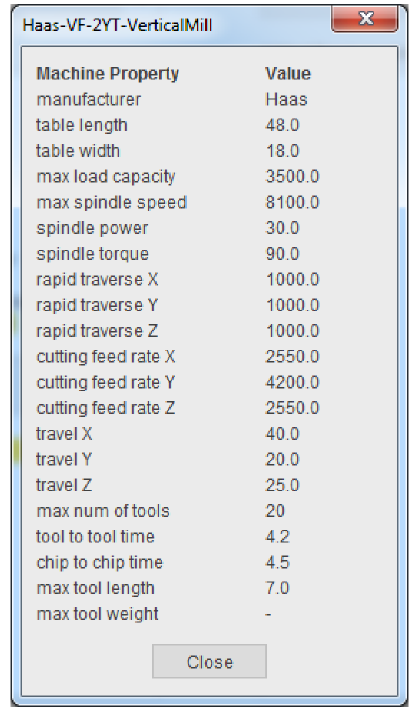
### 3.3.3 adding machines to the factory

1. **Select Equipment:** Select Machine Tool from the “Select Equipment” dropdown menu.

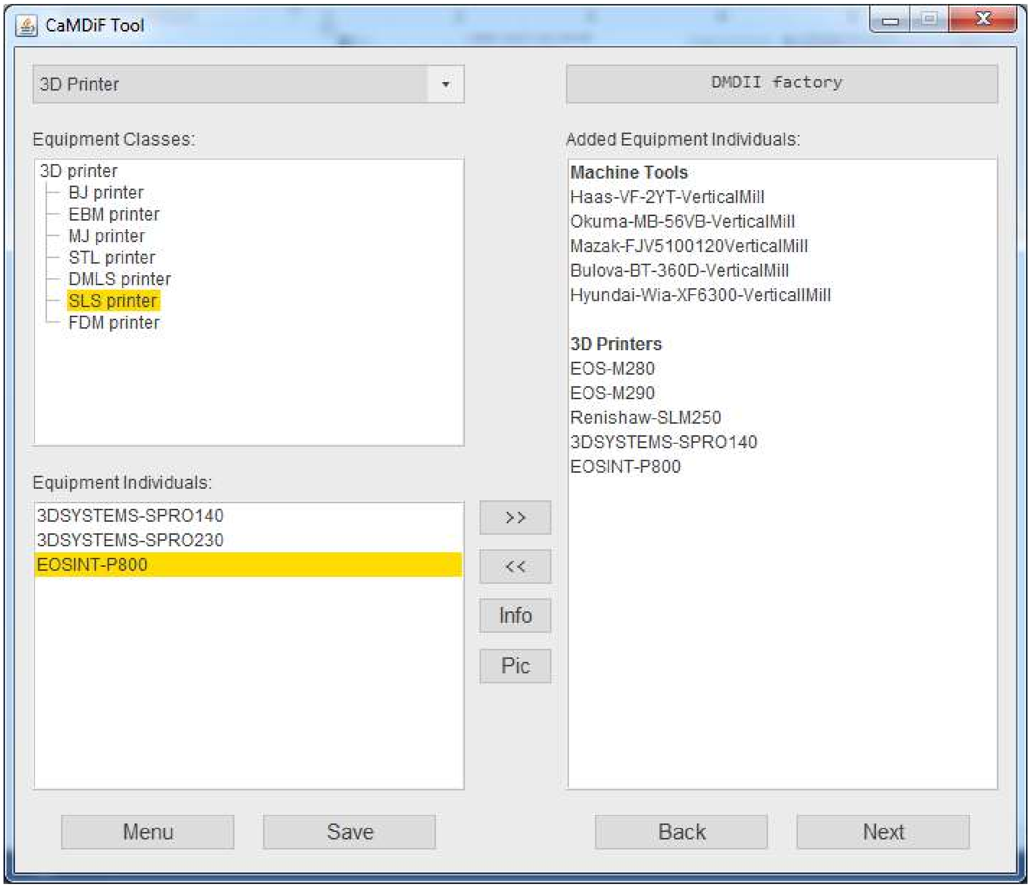



2. Select the type of equipment (for example Haas Vertical Mill) from the “Equipment Class” pane.
3. By selecting the equipment class, the lower pane is populated by specific equipment individuals from the selected class. Select the individual equipment you want to add to the factory and press the “>>” button.
    a. To delete equipment from the factory, select the equipment from the right pane and press “<<”.
    b. By clicking the “Info” button, the specifications of the selected equipment individual are shown in a popup screen.

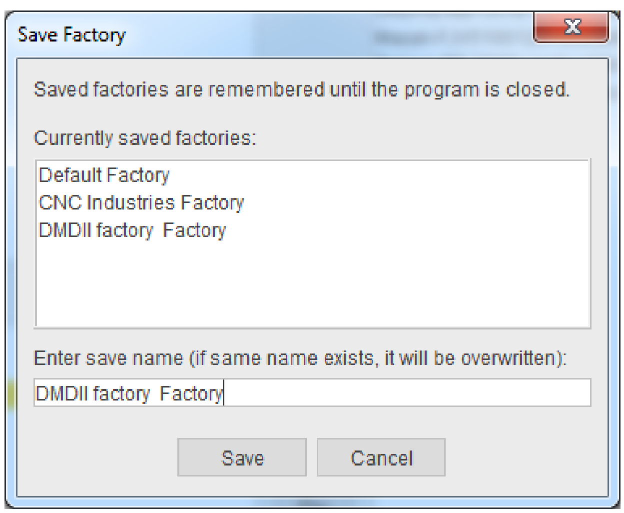



4. To add 3D printers, select “3D Printer” from the dropdown menu and follow the procedure described in step 3 (above).




5. **Save:** Once all machines are added to the factory, save the factory.



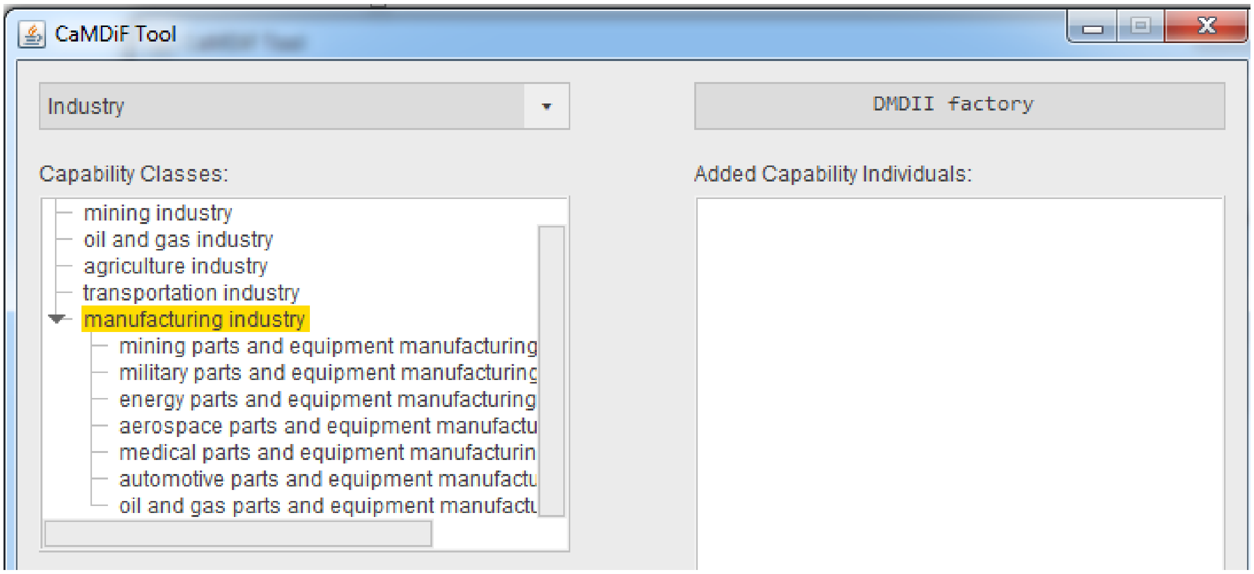

6. Press "Next" to go to the next step.

### 3.3.4 adding other capabilities to the factory

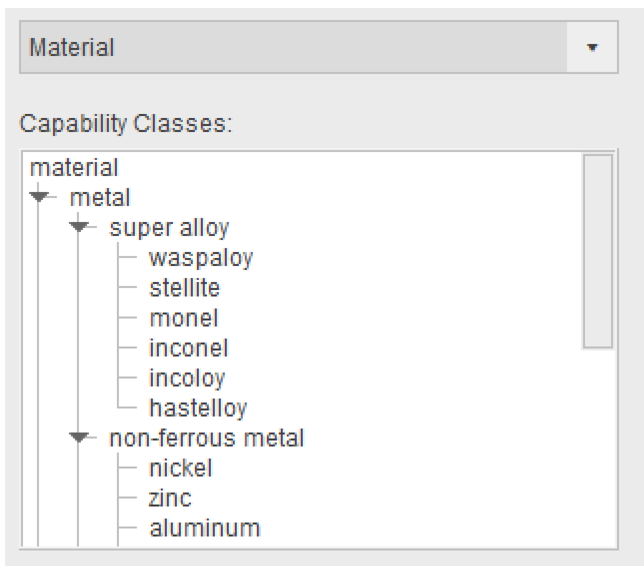
In this step, other types of capabilities (including industry, material, skill, and software) can be added.

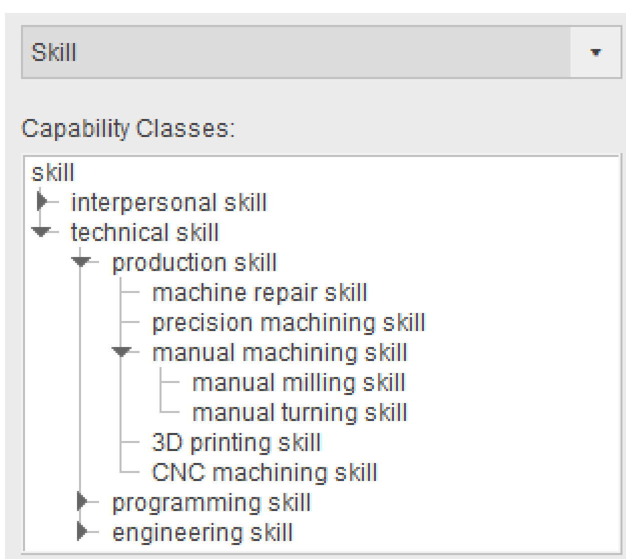
1. Select "Industry" from the dropdown menu.
    a. You will select the industries that the company typically serves.
2. Select the industry class from the top left pane.

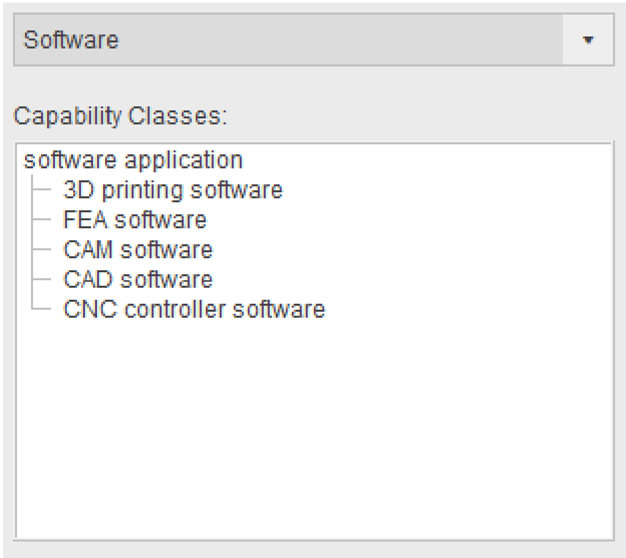
3. Select the industry individual pertaining to the selected class from the bottom pane.
4. Press “>>” to add the selected industry individual to the factory (right pane).



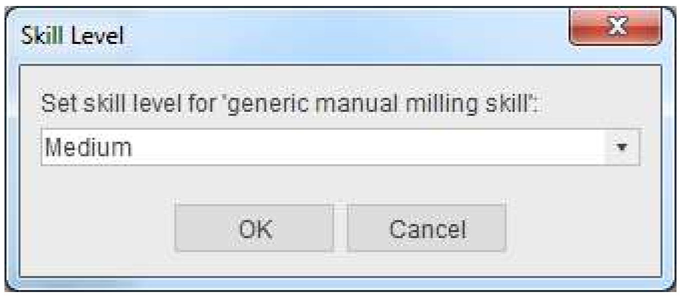

5. Repeat steps 2-4 for skill, material, and software.

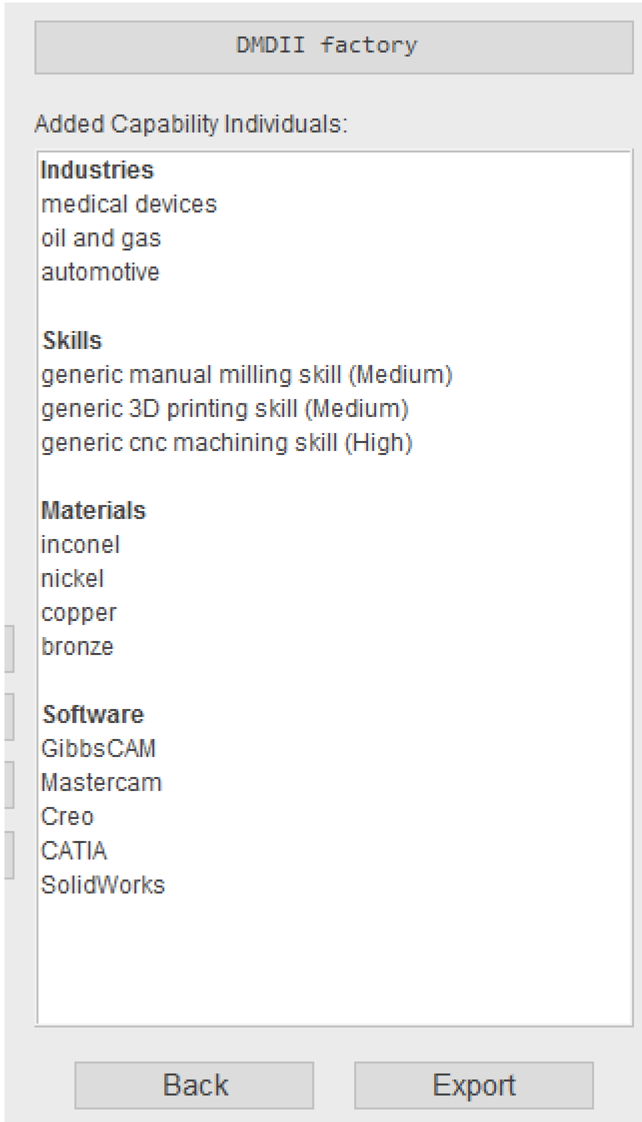


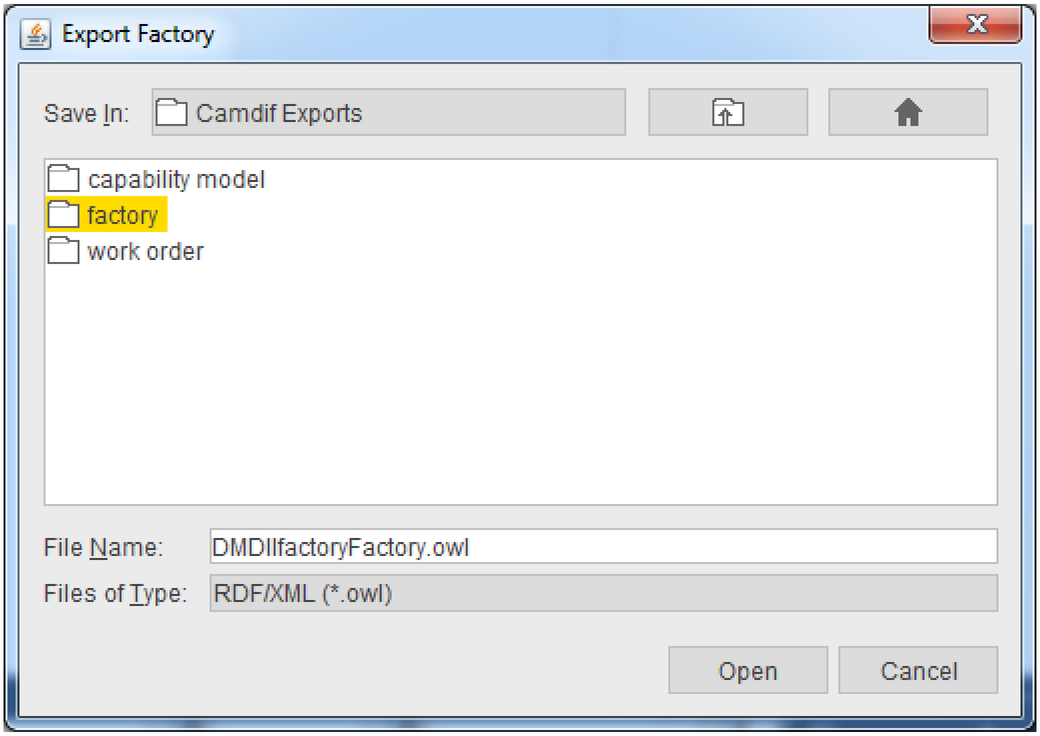



a. Note: When adding "skill" individuals, the level of skill (very low, low, medium, high, very high) should be given for each skill item.






6. **Save.**
7. **Export:** Export the factory file as an RDF/XML file on your local computer. You can always import the exported file for further processing.  But the saved files are only available in the current session.




This is the end of the Build module. To analyze the created factory, click “Menu”.

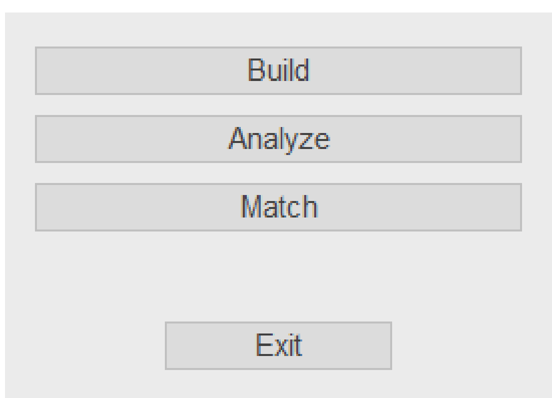
# ANALYZE

The main function of this module is to interpret the manufacturing capabilities of the user-made factories. Using this module, either a single factory is analyzed individually or two factories are compared with each other.

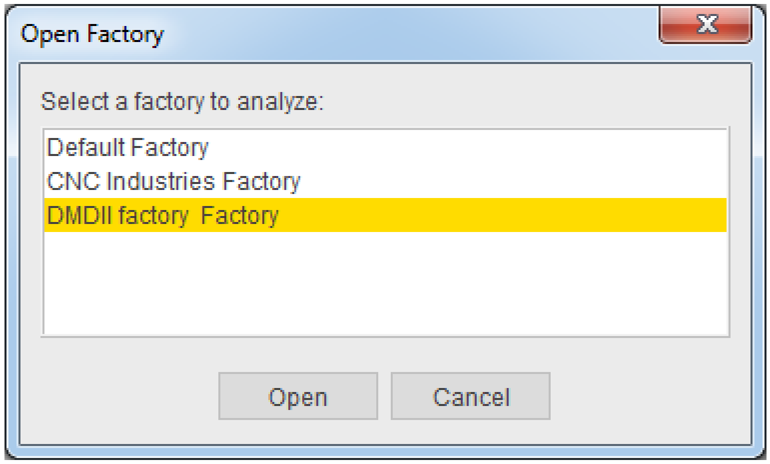
### 3.3.5 analyze factory (one factory)

The analysis begins with selecting an existing (saved) factory or importing an external factory previously created using the CaMDiF tool.

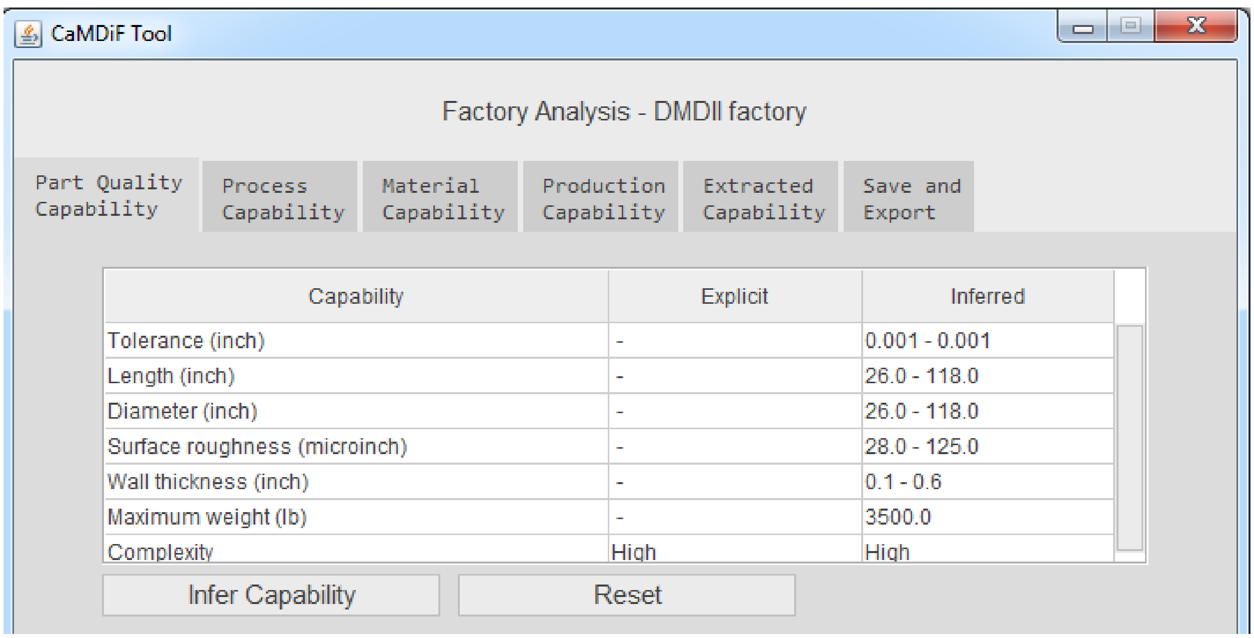
1. Select Analyze from the launch screen.



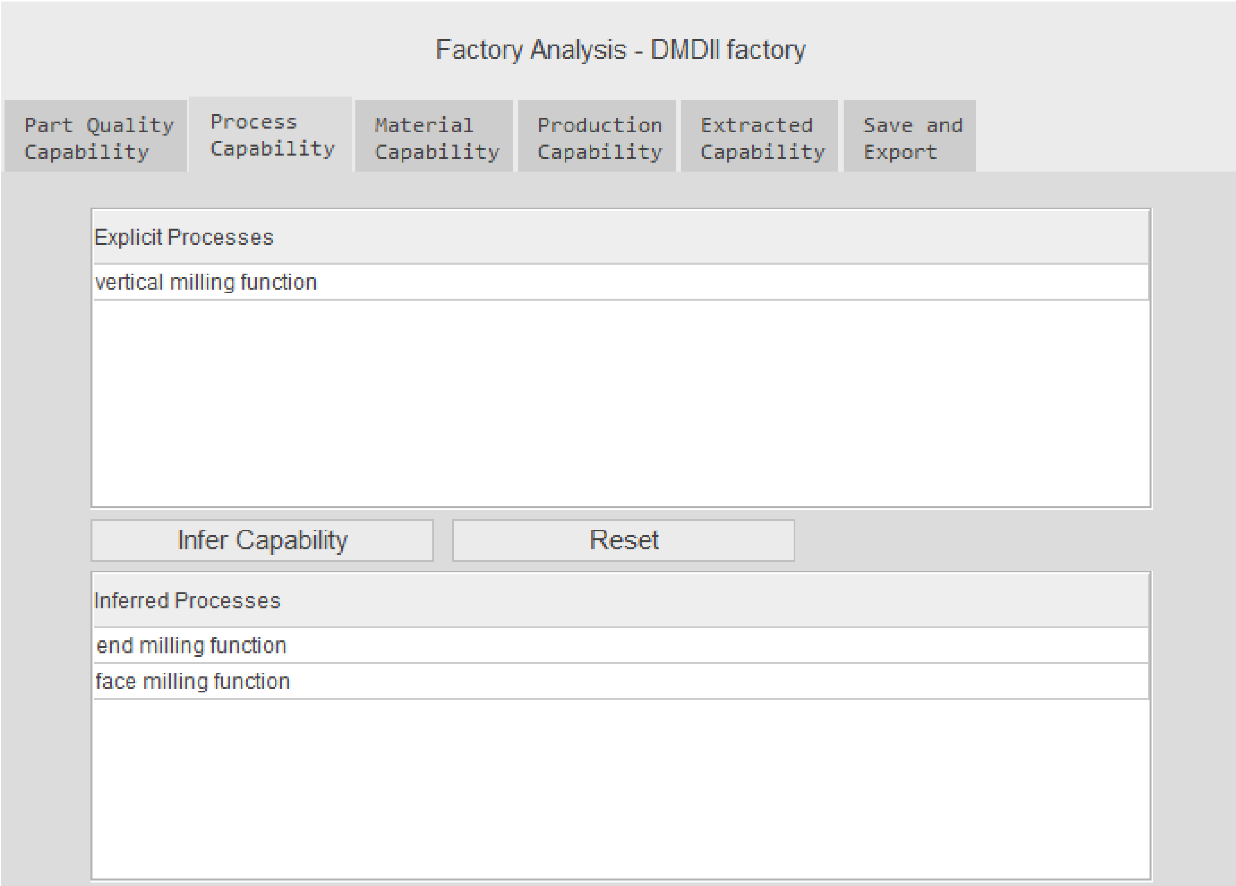

2. Select the factory you want to analyze.



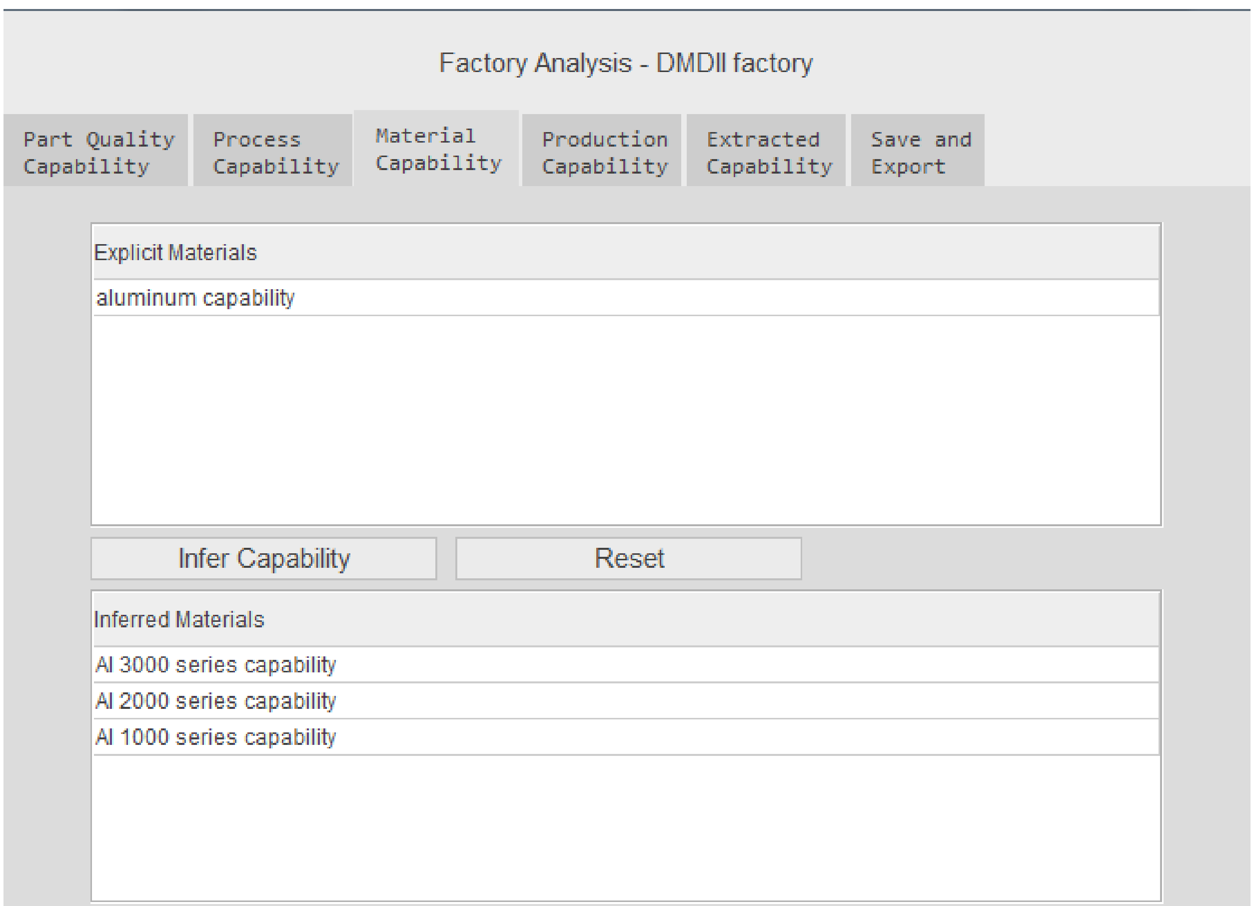

3. Part Quality Capability Tab: The first tab shows the capability of the factory with respect to the part attributes. If one or more cells in the “Explicit” column is empty, press “Infer Capability” to populate the inferred column.



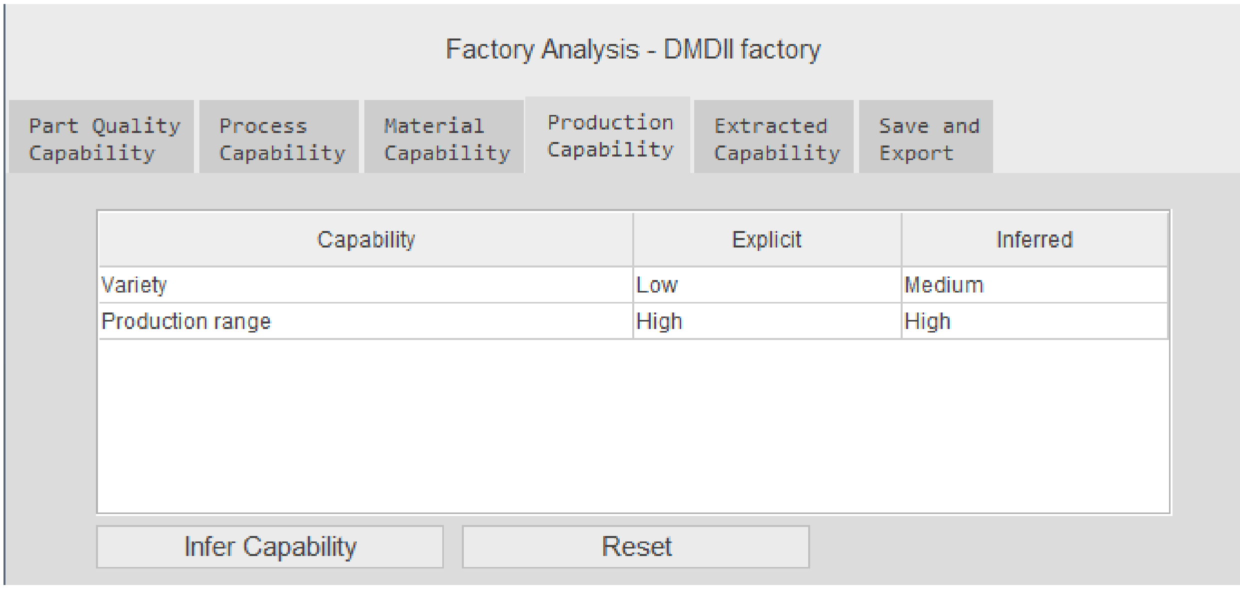

4. Process Capability Tab: This tab shows the types of manufacturing processes (explicit and implicit) available at the factory.




5. Material Capability Tab: This tab shows the list of materials that can be processed at the factory.

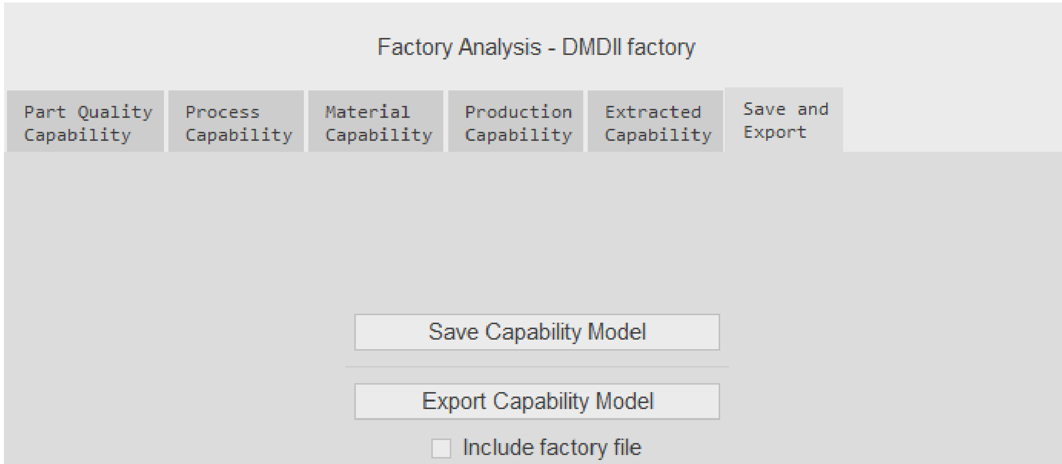


6. Production Capability Tab: This tab shows the production capability of the factory in terms of the variety of parts that can be manufactured, as well as production volumes.



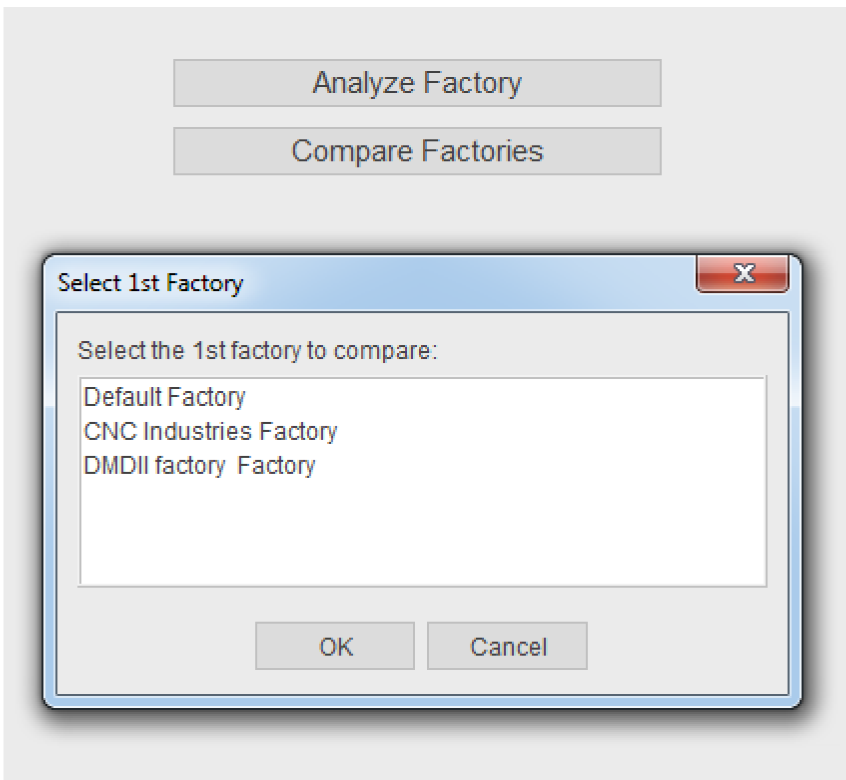

7. Extracted Capability Tab: This tab shows the keywords extracted from the website of the company categorized by different capability features. The number in the parenthesis next to a term shows the term's (and its synonyms') frequency of occurrence on the website's pages.

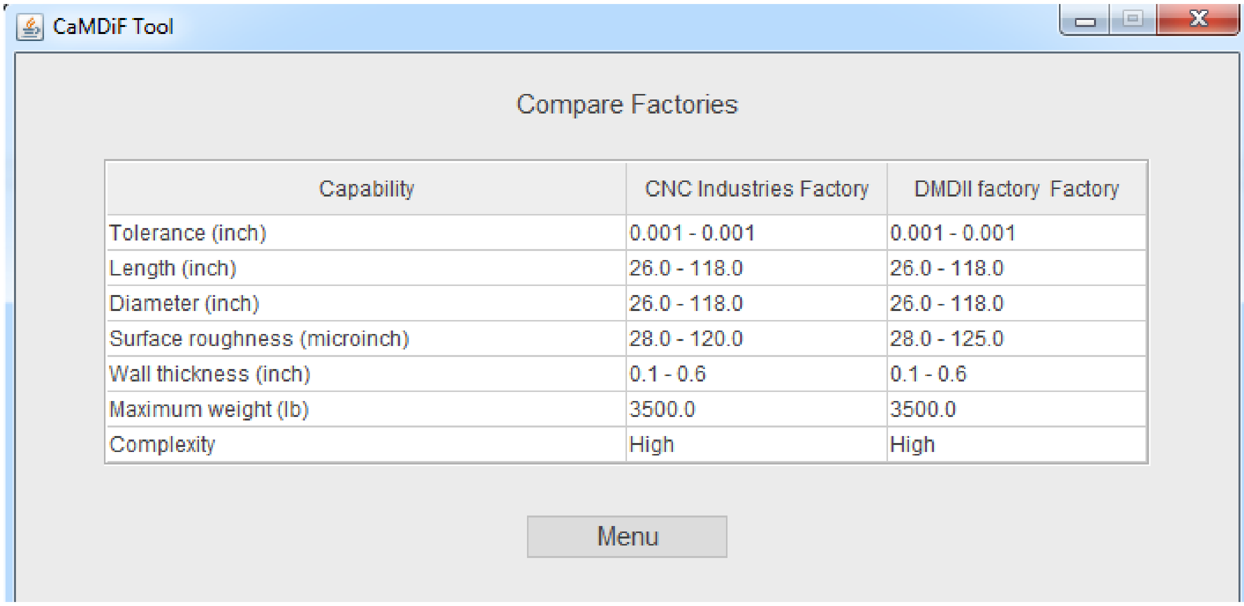

8. Save and Export: The capability file can be save and/or exported as an XML/RDF file. If the capability file is saved, then it is deleted once the session is closed. Exported files are saved permanently at a given location on the local drive. If "Include Factory File" is checked, then the capability model of the factory includes the factory file (machines, skills, software packages, etc) as well.




### 3.3.6 compare factories

1. Press Compare Factories
2. Select the first factory to compare
3. Select the second factory to compare






| Capability | CNC Industries Factory | DMDII factory Factory |
|---|---|---|
| Tolerance (inch) | 0.001 - 0.001 | 0.001 - 0.001 |
| Length (inch) | 26.0 - 118.0 | 26.0 - 118.0 |
| Diameter (inch) | 26.0 - 118.0 | 26.0 - 118.0 |
| Surface roughness (microinch) | 28.0 - 120.0 | 28.0 - 125.0 |
| Wall thickness (inch) | 0.1 - 0.6 | 0.1 - 0.6 |
| Maximum weight (lb) | 3500.0 | 3500.0 |
| Complexity | High | High |

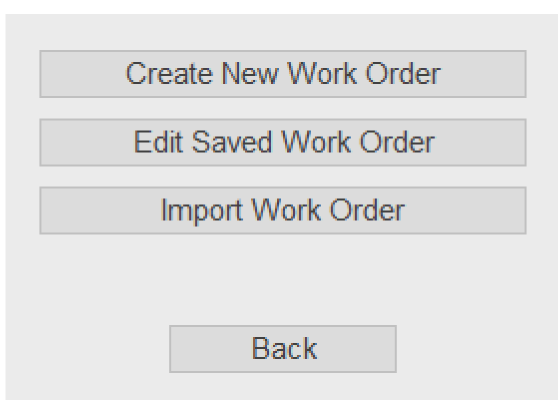
This is the end of the Analyze module. To run the Match module, click “Menu”.

# MATCH

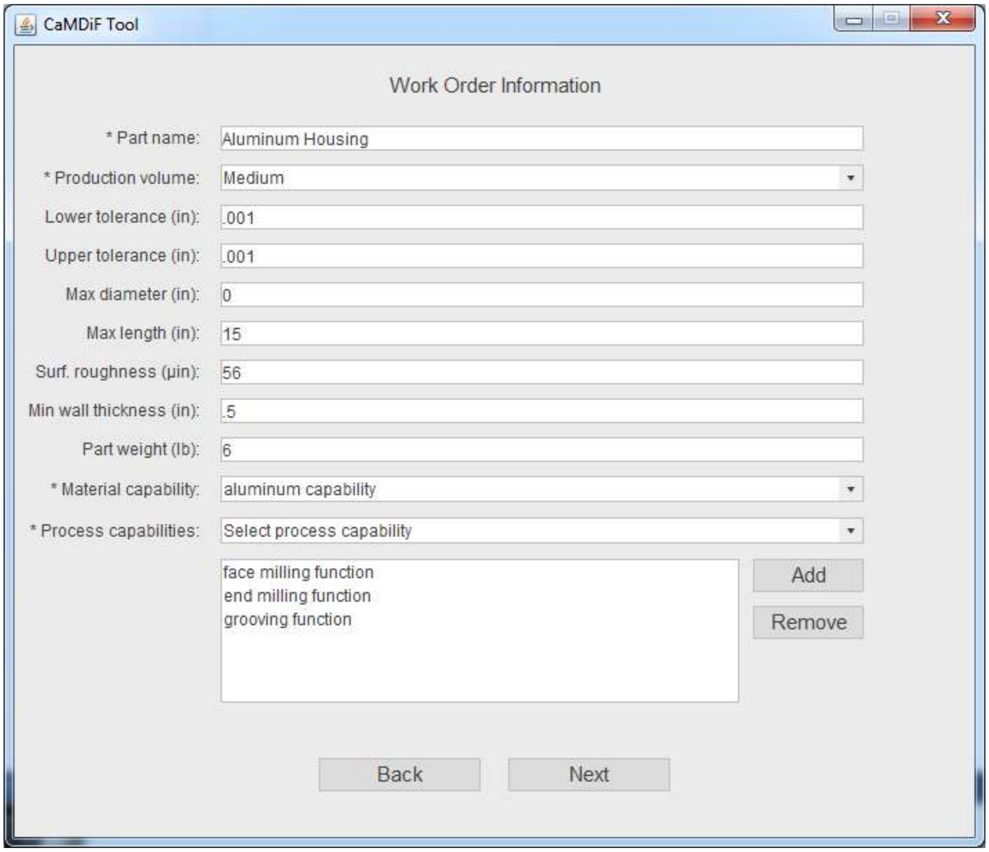
The match module is used for creating work orders and building supply chains that can manufacture the part described in the work order with the required quality and quantities.

### 3.3.7 create new work order

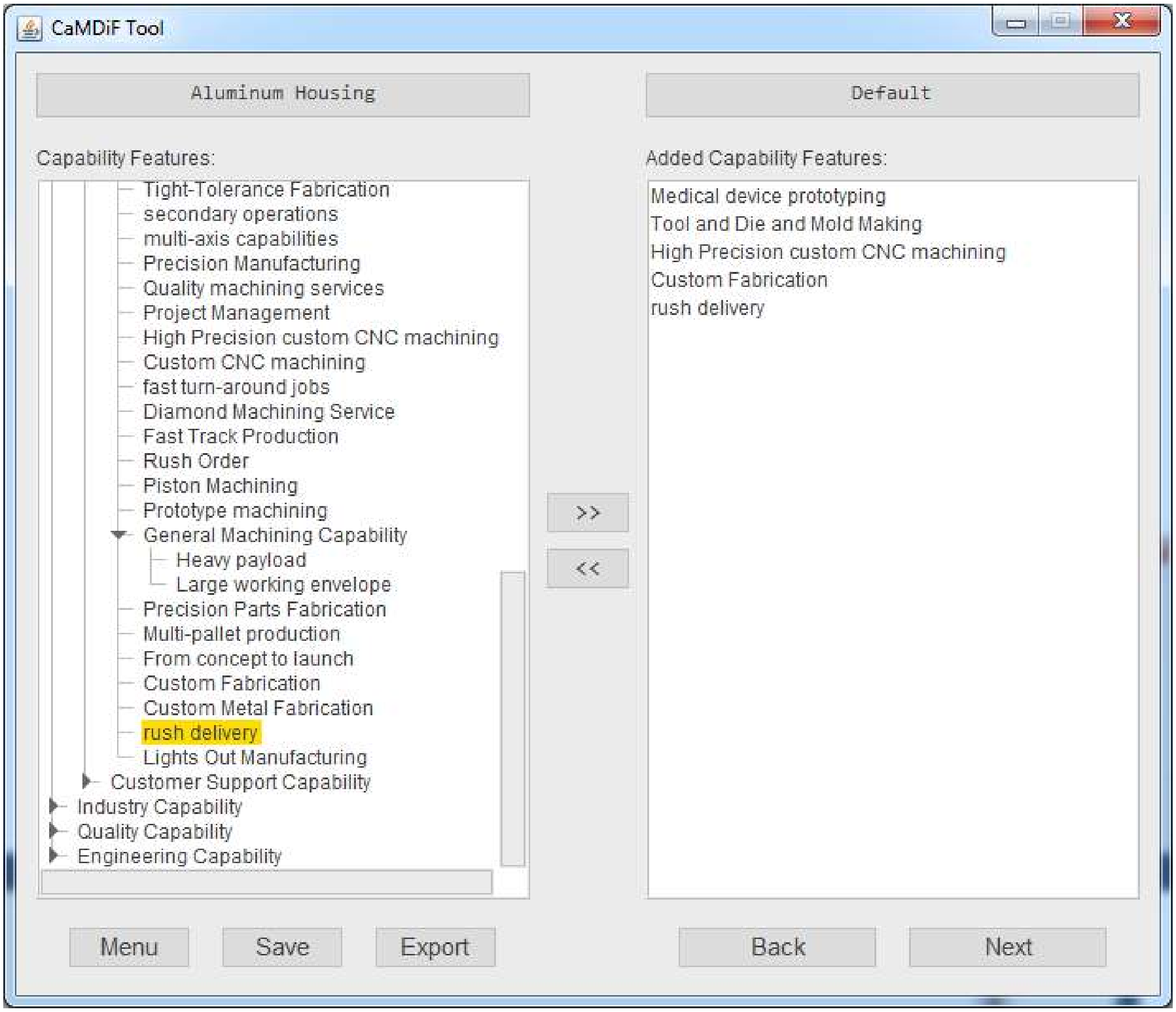
1. Select **Create New Work Order.**
2. Enter the company information (similar to the first step in the build module). The company created at this stage is the issuer of the work order.
    a. Note: Alternatively, you can open a saved/imported work order and then edit it.
    b. All imported work orders are saved automatically.



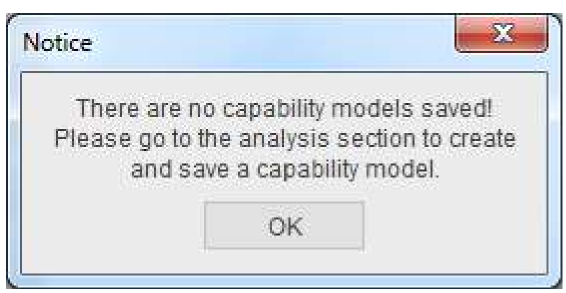

3. Enter the work order information (including part name, part materials, production volume, tolerances, dimensions, required processes, etc.).




4. Add desirable capability features by browsing the tree structure.
    a. Here, *feature* refers to an aspect of capability represented by a term or phrase such as *turnkey service*, or *heavy part machining.*
    b. These features will be used in the next step to optimize the generated supply chains.

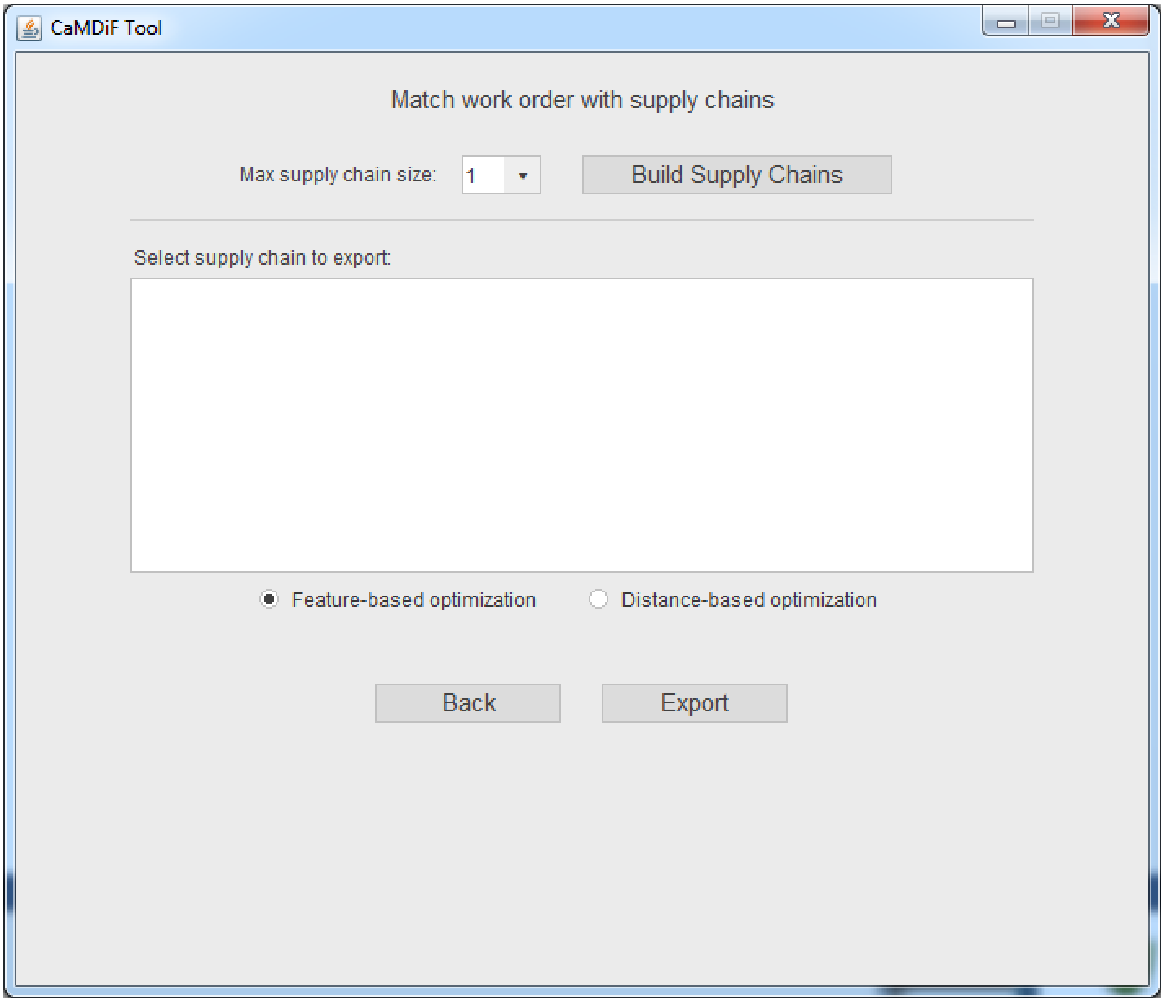



5. Save and/or export the work order.
6. Press “Next” to go to the next step (matching the work order with capability models of the saved factories)
7. If no capability model is saved in the “Analyze” step, an error message appears on the screen.




### 3.3.8 Match work order with supply chains

1. Select the max allowable size of the desirable supply chain (1-4).
2. Select type of optimization method (feature-based or distance-based)
3. Click “Build Supply Chains”


CaMDiF Tool
Match work order with supply chains
Max supply chain size:
1
Build Supply Chains
Select supply chain to export:
Feature-based optimization
Distance-based optimization
Back
Export