



Web-Based Graphics for Battlespace Terrain Ownership

by Andrew M. Neiderer

ARL-MR-648

August 2006

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Aberdeen Proving Ground, MD 21005-5067

ARL-MR-648**August 2006**

Web-Based Graphics for Battlespace Terrain Ownership

Andrew M. Neiderer

Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188		
<p>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) August 2006		2. REPORT TYPE Final		3. DATES COVERED (From - To) January–February 2006	
4. TITLE AND SUBTITLE Web-Based Graphics for Battlespace Terrain Ownership			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Andrew M. Neiderer			5d. PROJECT NUMBER P622783Y10		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory AMSRD-ARL-CI-CT Aberdeen Proving Ground, MD 21005-5067			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-MR-648		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <p>An approach for the display of battlespace terrain ownership data on a semantic Web is presented. Scaleable vector graphics (SVG), a two-dimensional Web graphic based on the extensible markup language (XML), is used. The SVG document accesses and modifies XML data using interpreter language ECMAScript variables and functions, as well as globally-defined objects of the document object model for the Batik SVG browser, from its <script> element. ECMAScript functions were written for dynamic updating on the client machine. Interactive visualization of the scene-graph using extensible three-dimensional graphics, which is also XML-based, is being added for user inputs and external events.</p>					
15. SUBJECT TERMS scaleable vector graphics, extensible markup language, battlespace terrain ownership					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UL	18. NUMBER OF PAGES 22	19a. NAME OF RESPONSIBLE PERSON Andrew M. Neiderer
a. REPORT UNCLASSIFIED	b. ABSTRACT UNCLASSIFIED	c. THIS PAGE UNCLASSIFIED			19b. TELEPHONE NUMBER (Include area code) 410-278-3203

Contents

1. Introduction	1
2. SVG Web Graphic for BTO Display	2
3. X3D Entities	4
4. Conclusion and Future Efforts	5
Appendix. Sample SVG Document and Update	7
Distribution List	14

INTENTIONALLY LEFT BLANK.

1. Introduction

The saying that a picture is worth a thousand words implies that humans are fundamentally better at the processing of data through illustration than in a raw literary form, especially when dealing with large volumes of data. Now with the emergence of a semantic Web, the use of a style sheet with the extensible markup language (XML) not only presents the data but describes it as well. The result is two-dimensional (2-D) and three-dimensional (3-D) Web graphics, where currently the languages of choice are scalable vector graphics (SVG) and extensible 3-D (X3D), respectively, that can enhance the decision-making process. An example may be a Commander on the battlefield who wants to quickly visualize critical data before selecting a course of action. The Battlespace Decision Support Team (BDST) at the U.S. Army Research Laboratory (ARL) is developing 2-D Web graphics for such a tool: its battlespace terrain ownership (BTO) system.

This report discusses Web-based graphics for BTO.¹ The objective is to provide an accurate representation of current conditions and to maximize an understanding so that decisions can be made quickly. BTO does this by displaying expected control of battlespace areas over time, based on position, influence exerted by asset distribution, weapon system effectiveness, probabilities of hit and kill, and combat damage. Six classifications of ownership ratios are defined for the blue (B) and red (R) forces on a battlefield (see table 1). The seventh and final classification designates any “zero” owned areas, i.e., neither force has any influence due to weapon range restrictions.

Table 1. Classification of BTO data.

Classification	Color	Hexadecimal Code
B:R \geq 6:1	Strong blue	#151B8D
3:1 \leq B:R < 6:1	Medium blue	#2B60DE
1:1 \leq B:R < 3:1	Weak blue	#3BB9FF
1:1 \leq R:B < 3:1	Weak red	#F62817
3:1 \leq R:B < 6:1	Medium red	#F75D59
R:B \geq 6:1	Strong red	#FF0000
Zero owned	Green	#00FF00

Note the assignment of a hexadecimal value for a given color. This is because color interpretation by humans is not necessarily unique, i.e., different combinations of red-green-blue (RGB) components can result in the same sensation (see description of metamers by Foley et al.²). We eliminate this many-to-one mapping by specifying the actual RGB color components used in our SVG document.

¹O'May, J.; Hansen, C.; Heilman, E.; Kaste, R.; Neiderer, A. Battlespace Terrain Ownership: A New Situation Awareness Tool. *Proceedings of the 10th International Command and Control Research and Technology Symposium*, McLean, VA, 13–16 June 2005.

²Foley, J.; van Dam, A.; Feiner, S.; Hughes, J. *Computer Graphics Principles and Practice*; 2nd ed.; Addison-Wesley Publishing Company, Inc.: Reading, MA, 1990.

2. SVG Web Graphic for BTO Display

This section describes in detail how we design an SVG scene-graph to visualize BTO data. The goal is to quickly provide the military Commander a best estimate of current expected control of the different areas on a changing battlefield. We also want to maximize human understanding of this data and display patterns and relationships for insightful decision-making. The result is an effective, yet efficient, display that should increase the situational awareness of a Commander and his troops and contribute to their success.

An effective display of data maps physical representations to perceptual relationships. This is the approach we use when displaying ownership data for different scenarios using the One Semi-Automated Forces (OneSAF) simulation. An example frame of ownership is given in figure 1. A transparency, where $\alpha = 0.4$ on the unit interval ranging from transparent to opaque, of the image is accurately registered with level one digital terrain elevation data (DTED-1) and displayed using Java3D³ (see figure 2). Note that the darker areas in this figure are due to the angle between a directional light source and the selected viewing position.

The SVG for this rendering was first pre-authored in Java⁴ for the Apache Software Foundation Batik viewer (<http://xml.apache.org/batik/>). It scripts the document object model (DOM) using the ECMAScript interpreter language variables and functions, and globally defined objects for the viewer. The DOM is an abstract interface that allows for manipulation of the scene graph. Each browser defines global objects for a particular instance. We selected viewing our SVG document with Batik, and therefore access and modify exposed elements and attributes for that viewer. This can be done for either a Netscape, Inc. or a Microsoft, Inc. Web browser, but SVG plug-in software must first be installed.

An “*id*” and “*class*” attribute were assigned to each rectangle (i.e., `<rect>`). The *onload* event handler is executed when the document is loaded by placing it in the `<svg>` root element. This starts the *startBTOdisplay()* method, which initializes the *setInterval()* of the Window object for the viewer. An interval time is assigned for a frame update of at least 60 Hz, or 30 frames per second, which is the recommended refresh rate by the National Television System Committee (NTSC) for frame-based rendering. The *updateFrame()* method then requests data from the client using *getURL()*.^{*} A parameter of this method is the callback function, which then checks for updates at the desired frequency. The appendix includes full source code.

³Sun Microsystems, Inc. <http://java3d.dev.java.net> (accessed October 2004).

⁴Sun Microsystems, Inc. <http://java.sun.com> (accessed August 2004).

^{*}The Batik SVG viewer does not support the standard XMLHttpRequest method used in some browsers, like Microsoft, Inc. Internet Explorer and the Mozilla, Inc. Firefox Web browser. However, it does provide the same functionality.

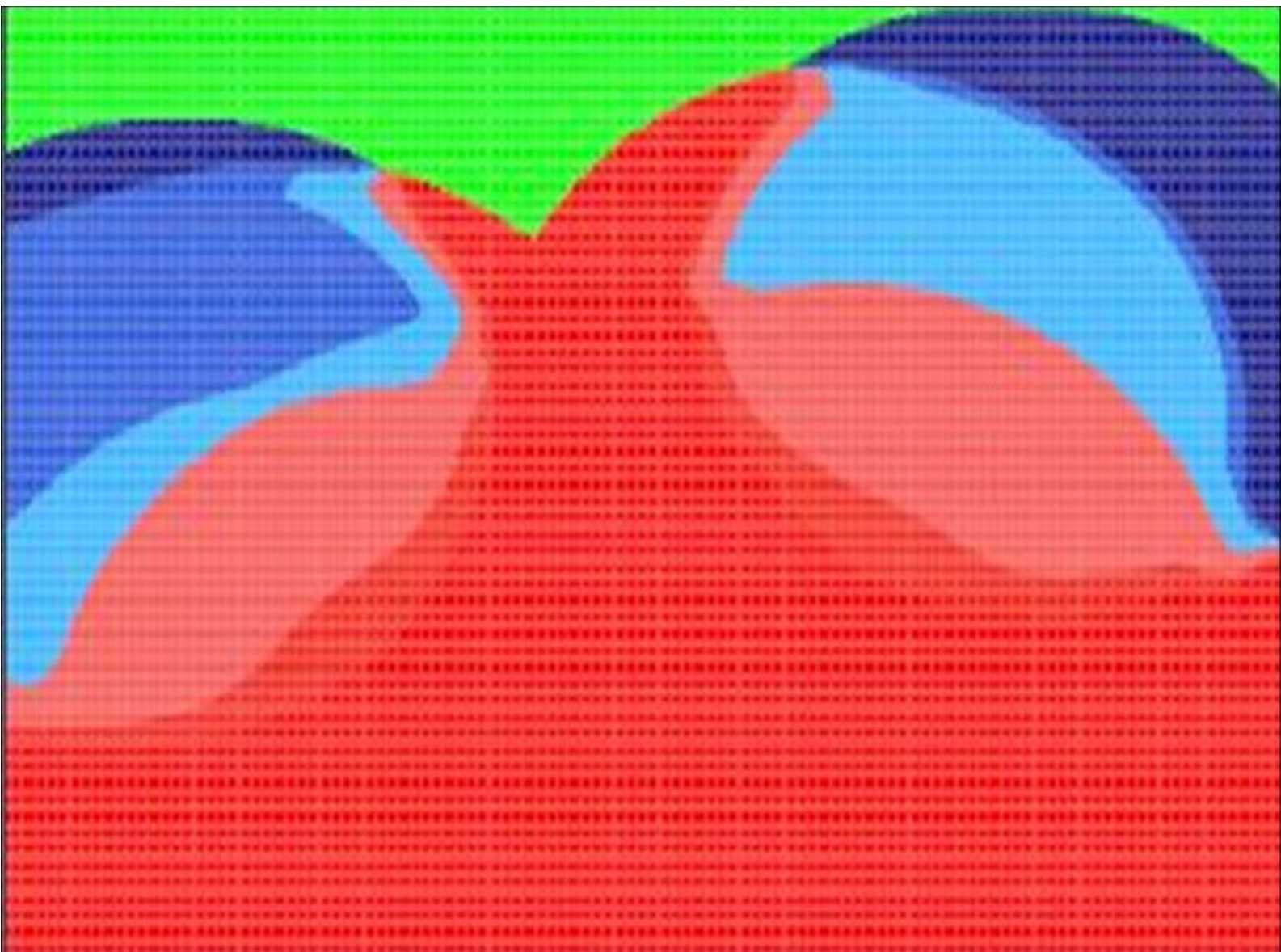


Figure 1. Example of frame ownership.

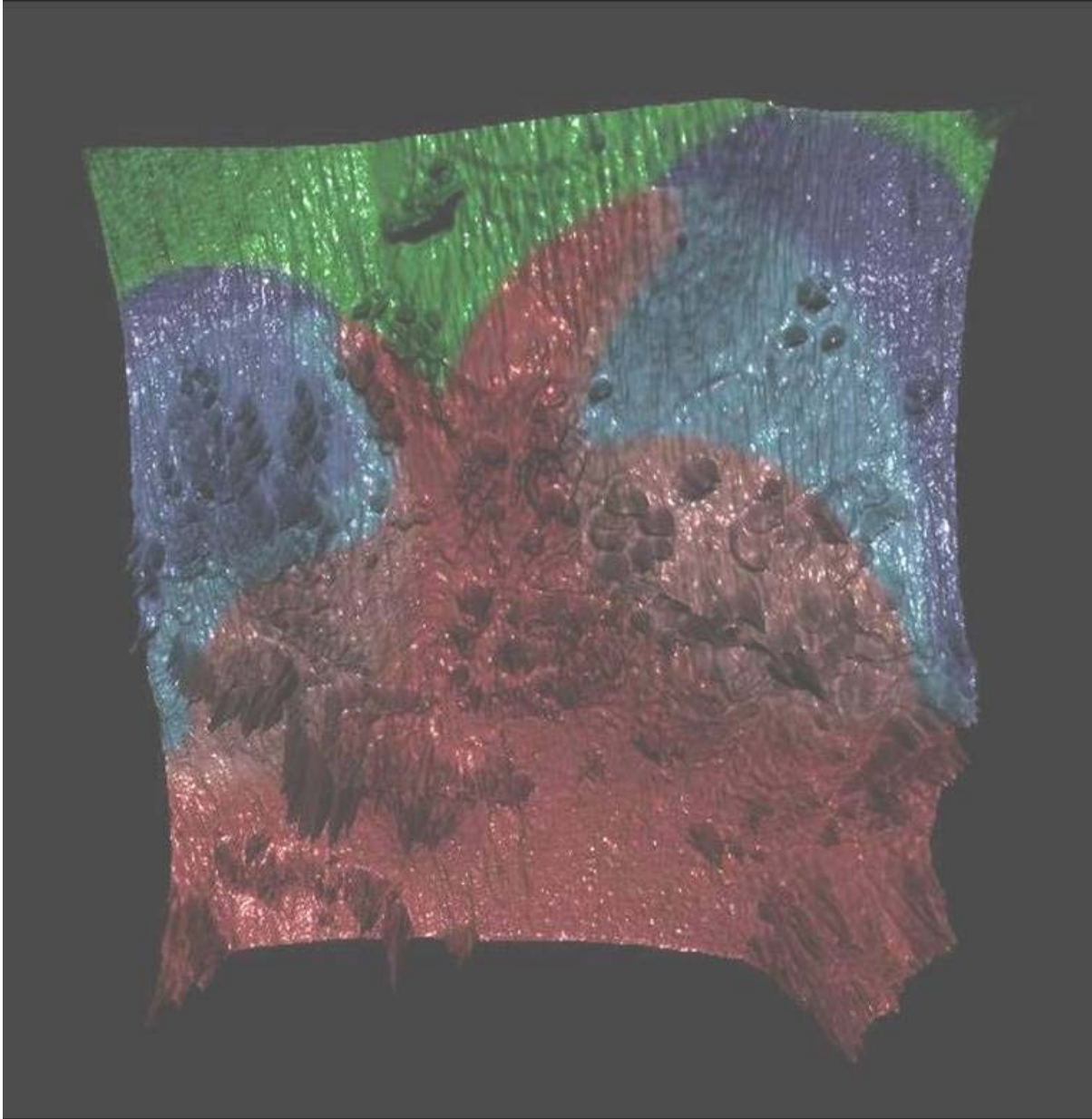


Figure 2. Transparency image accurately registered with DTED-1 and displayed using Java3D.³

3. X3D Entities

An ECMAScript handler for *onmouseover* events within the Batik viewer is being written for identifying entities within the active battle area. Both entity identification and position are available for a particular OneSAF scenario, and the screen size is defined by the user in the `<svg>` root element. Thus, placement of entities will involve a linear mapping from the user coordinates of OneSAF to the device space of our window.

An X3D description of entities will also be added to the BTO system. X3D, like SVG, is an XML encoding and the successor to the virtual reality modeling language (VRML). It improves the communication among <Scene> elements distributed over a network of nodes. The result is a more interactive scene graph capable of user inputs and external events.

The interactive virtual environment (VE) is a directed acyclic graph of objects in the 3-D world, and the relationships among the objects (i.e., the transformation and behavior hierarchy when constructing the <Scene>). The VE will be fully immersive with navigation and environmental sensor control. Interactive content in X3D is achieved by accessing and modifying field values of sensor elements, and then *ROUTEing* these changes to interpolator or <Script> elements. The interface between X3D scene content and Java member data and functions, as well as any globally-defined objects for the DOM of the Xj3D browser (<http://www.xj3d.org>) in our case, are done from this <Script> node of the X3D <Scene>. In other words, the scene access interface (SAI) from the <Script> node is the set of interface classes to scene content from a Java class. The passing of messages from one node to another along a defined *ROUTE* constitutes event propagation.

Note that the Naval Postgraduate School (Monterey, CA) has developed a large collection of 3-D military models in their Scene Authoring and Visualization for Advanced Graphical Environments (SAVAGE) project. These were written in X3D and will be evaluated for inclusion.

4. Conclusion and Future Efforts

SVG has been written for the rapid visualization of ARL's BTO. The SVG document scripts the DOM of the Apache Batik viewer in ECMAScript when determining the 2-D graphic of terrain ownership. An X3D script for the DOM of the Xj3D browser is being written for 3-D display of entities within the battle area as determined by BTO. Presently, an "all-in-one" browser for mixing 2-D and 3-D XML data for visualization does not exist, but integration will be attempted for SVG and X3D.

Also, the BDST at ARL is extending their BTO to an urban environment. Military operations on urban terrain are difficult to simulate. For example, people and structures are just two of the many entities that may have to be avoided. And assuming that buildings are present, the determination of ownership may be difficult since the possibility exists that it can be destroyed at any time. There are many other complications, so a statistical determination may be necessary.

But the graphics programming modification should not be that difficult. The OneSAF Objective System, which is being written in Java, has a requirement that its environmental data model for entity level simulation defines ultra-high resolution buildings (UHRB) in XML. Thus, X3D, and hopefully SVG, will likely be used for the entire scene-graph construction.

INTENTIONALLY LEFT BLANK.

Appendix. Sample SVG Document and Update

Scaleable vector graphics (SVG) markup and ECMAScript for document object model change of the Batik SVG viewer used in the example throughout this report are now included. XML data is accessed and modified in our document using the “*id*” and “*class*” attributes of a rectangle `<rect>` element. The area for determining ownership is roughly $23,600 \times 12,700$ m, and the device space is 784×422 pixels. This computes to about 30 m pixels. Note that only a partial listing of `<rect>`s is included due to space restrictions.

A sample update is then included. Modifying the SVG document in this manner is faster than a total update for the next frame, assuming that not much changes from frame to frame.

```

<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1/EN"
    "http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
<svg xmlns='http://www.w3.org/2000/svg'
    xmlns:xlink='http://www.w3.org/1999/xlink'
    viewBox='0 0 23600 12700'
    width='784' height='422' preserveAspectRatio='none'
    onload='startBTODisplay()'>

<desc>
  <title>
    <!-- Battlespace Terrain Ownership visualization using Scaleable Vector    -->
    <!-- Graphics (SVG), ie a 2D Web graphic based on XML technology. Also    -->
    <!-- scripts the document object model (DOM) for the Batik SVG Viewer    -->
    <!-- using ECMAScript for rapid update of the XML data on the client side. -->
    <!--                                     -->
    <!-- by Andrew M. Neiderer, 21 Jan 2006.                                     -->
  </title>
</desc>

```

```

∞ <defs>

  <!-- style sheet -->

  <style type='text/css'>
    <![CDATA[
      .strongBlue {fill:#151B8D;}
      .mediumBlue {fill:#2B60DE;}
      .weakBlue {fill:#3BB9FF;}
      .strongRed {fill:#FF0000;}
      .mediumRed {fill:#F62817;}
      .weakRed {fill:#F75D59;}
      .green {fill:#00FF00;}
      .white {fill:#FFFFFF;}
      .black {fill:#000000;}
    ]]>
  </style>

  <!-- scripting definitions -->

  <script type='text/ecmascript'>

```

```

<![CDATA[
  <!-- member data -->

  var frameUpdate;
  var frameUpdateStringName = 'frameUpdate';
  var frameUpdateStringExtension = '.txt';
  var frameUpdateNumber = 1;

  var intervalTime = 1;

  <!-- member functions -->

  function updateCallback(result)
  {
    var updates = result.content.replace(/\n/g,"").split(":");

    for ( var i in updates ) {
      var tmp = updates[i].split(",");

      var id = tmp[0];
      var eltID = document.getElementById(id);

      if ( eltID != null ) {
        var cls = tmp[1];
        eltID.setAttributeNS(null,"class",cls);

        var fillOpacity = tmp[2];
        eltID.setAttributeNS(null,"fill-opacity",fillOpacity);

        var animate =
          eltID.getElementsByTagNameNS("http://www.w3.org/2000/svg",
            "animate").item(0);

        if ( animate != null ) {
          alert(fillOpacity);
          animate.setAttribute("to",fillOpacity);
        }
        else
          alert("animate is null" + frameUpdateStringName +
            frameUpdateNumber + frameUpdateStringExtension);
      }
    }
  }
]

```

```

    }

    function updateFrame()
    {
        if ( frameUpdateNumber < 19 ) {
            frameUpdate = frameUpdateStringName + frameUpdateNumber +
                          frameUpdateStringExtension;
            getUrl(frameUpdate,updateCallback);
            frameUpdateNumber++;
        }
    }

    function startBTODisplay()
    {
        window.setInterval('updateFrame()',intervalTime);
    }
}]]>
</script>

```

</defs>

```

<rect id='r0c0' x='0.0' y='0.0' width='92.1875' height='49.609375' class='green' fill-opacity='0.05'/>
<rect id='r0c1' x='92.1875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c2' x='184.375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c3' x='276.5625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c4' x='368.75' y='0.0' width='92.1875' height='49.609375' class='green' fill-opacity='0.05'/>
<rect id='r0c5' x='460.9375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c6' x='553.125' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c7' x='645.3125' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c8' x='737.5' y='0.0' width='92.1875' height='49.609375' class='green' fill-opacity='0.05'/>
<rect id='r0c9' x='829.6875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>
<rect id='r0c10' x='921.875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05'/>

```



```

    <rect id='r0c11' x='1014.0625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c12' x='1106.25' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c13' x='1198.4375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c14' x='1290.625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c15' x='1382.8125' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c16' x='1475.0' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c17' x='1567.1875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c18' x='1659.375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c19' x='1751.5625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c20' x='1843.75' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
II  <rect id='r0c21' x='1935.9375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c22' x='2028.125' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c23' x='2120.3125' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c24' x='2212.5' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c25' x='2304.6875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c26' x='2396.875' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c27' x='2489.0625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c28' x='2581.25' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c29' x='2673.4375' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />
    <rect id='r0c30' x='2765.625' y='0.0' width='92.1875' height='49.609375' class='green' fill-
opacity='0.05' />

```

```
<rect id='r0c31' x='2857.8125' y='0.0' width='92.1875' height='49.609375' class='green' fill-  
opacity='0.05' />  
<rect id='r0c32' x='2950.0' y='0.0' width='92.1875' height='49.609375' class='green' fill-  
opacity='0.05' />  
.  
.  
.  
</svg>
```

frameUpdate15.txt

r86c91,weakBlue,0.20:r86c92,weakBlue,0.20:r86c93,weakBlue,0.20:r86c94,weakBlue,0.20

NO. OF
COPIES ORGANIZATION

1 DEFENSE TECHNICAL
(PDF INFORMATION CTR
ONLY) DTIC OCA
8725 JOHN J KINGMAN RD
STE 0944
FORT BELVOIR VA 22060-6218

1 US ARMY RSRCH DEV &
ENGRG CMD
SYSTEMS OF SYSTEMS
INTEGRATION
AMSRD SS T
6000 6TH ST STE 100
FORT BELVOIR VA 22060-5608

1 INST FOR ADVNCD TCHNLGY
THE UNIV OF TEXAS
AT AUSTIN
3925 W BRAKER LN
AUSTIN TX 78759-5316

1 DIRECTOR
US ARMY RESEARCH LAB
IMNE ALC IMS
2800 POWDER MILL RD
ADELPHI MD 20783-1197

3 DIRECTOR
US ARMY RESEARCH LAB
AMSRD ARL CI OK TL
2800 POWDER MILL RD
ADELPHI MD 20783-1197

ABERDEEN PROVING GROUND

1 DIR USARL
AMSRD ARL CI OK TP (BLDG 4600)

NO. OF
COPIES ORGANIZATION

ABERDEEN PROVING GROUND

8 DIR USARL
 AMSRD ARL CI CT
 P JONES
 R KASTE
 A NEIDERER (5 CPS)
 M THOMAS

INTENTIONALLY LEFT BLANK.