

Networthiness Assessment Determination Checklist

Purpose: This document outlines the process and provides a checklist for determining the applicability and depth of networthiness assessments performed on new and modified/upgraded Information Technology (IT) systems and applications (capabilities).

Applicability: Each new IT capability and all capability modifications/upgrades must be assessed to verify/validate its Networthiness (network security, network impact, compatibility with the infrastructure, infrastructure requirements, spectrum support, security policy compliance, JTA/i-TRM standards compliance, communications and information manpower, training, logistics support, schedule, and funding). (25 Apr 02 SAF/AQ, AF C4ISP Policy memo).

Information Superiority. The Defense acquisition community shall provide U.S. forces with systems and families of systems that are secure, reliable, interoperable, and able to communicate across a universal information technology infrastructure, including National Security Systems, consisting of data, information, processes, organizational interactions, skills, analytical expertise, other systems, networks, and information exchange capabilities. (30 Oct 2002 DepSecDef Memo, *Defense Acquisition*)

Desired Result:

Determination whether the new IT capability or modification/upgrade does or does not require a Networthiness Assessment and issuance/re-issuance of a Certificate of Networthiness.

Warning to Program Managers and IT Capability Owners: Failure to properly identify to the assigned AFCA Action Officer (AO) systems/applications and associated modifications/upgrades which require Networthiness Assessments could stop, delay, or revoke the IT capability's ability to receive or retain a Certificate to Operate (CtO) and base connection (local DAA) approval. PMs and functional owners must contact the assigned AFCA AO if there are problem areas or questions relating to the networthiness assessment process.

Definitions :

IT Capability: An information technology system or software application.

Introduction: There are six sections that may be completed. Sections 1 through 4 must be completed for all circumstances. Complete either section 5 or 6 as applicable.

- Section 5 for new IT capabilities or capabilities which have not been issued a Certificate of Networthiness (CoN).
- Section 6 for modifications and upgrades to IT capabilities previously issued a CoN.

=====

Required Inputs for Determining Whether a new IT capability or Modification/Upgrade Requires a Certificate of Networthiness

Section 1. IT Capability

Name: Content Management System – Phase I

Acronym: CMS

Version Number: Phase 1.3 - Information Management Tool (IMT) Viewer 6.0.0 and IMT Designer 2.3.0

Summary of Changes: The CMS IMT Viewer and IMT Designer commercial-off-the-shelf products have been modified under new releases as summarized below (from PureEdge ICS Viewer and ICS Designer release information July 2003).

1.0 Viewer 6.0.0 Modifications

1.1 XFDL Changes

The Viewer implements all of the XFDL changes listed in this section.

1.1.1 Namespace Support

Namespace support has been added to the XFDL language for XFDL documents version 6.0 and higher. Note that:

- The XFDL namespace URL is: `http://www.PureEdge.com/XFDL/6.0`
- The XFDL namespace is the default namespace for the document.
- All non-XFDL elements must have their own namespace.
- You must declare the default namespace, xfdl namespace, and any custom namespaces at the beginning of your XFDL form, immediately following the `<?xml version="x.x"?>` tag.

For example:

```
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
xmlns:xfdl="http://www.PureEdge.com/XFDL/6.0"
xmlns:custom="http://www.PureEdge.com/XFDL/Custom">
```

1.1.2 Defining a Form's Version

With the introduction of namespace support, a form's XFDL version is now defined by the first namespace declaration on the XFDL tag. For example, a version 6.0 form will include the following XFDL tag:

```
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
```

However, you can still reference a form's version as though it were a global option, as shown:

global.global.version

1.1.3 XML Data Model

XFDL now supports an XML Data Model. This model allows form designers to create separate blocks of arbitrary XML within an XFDL form that share data with form elements, such as fields. This is useful for integrating with applications that require data in a particular XML format, such as schema-compliant data. The XML Data Model is based on the XForms standard, as published by the W3C, but is not limited to XForms. Refer to the XFDL Specification for more information.

1.1.4 XFDL Computes

XFDL computes are now attributes of the option tag. As a result, computes are contained in the opening tag of the option, and the current value of the compute is stored as simple character data in the option. The compute itself is enclosed by quotes. For example:

```
<field sid="FIELD1">
  <value compute="PAGE1.nameField.value">Jane E. Smith</value>
</field>
```

Additionally, because computes are now enclosed in quotation marks ("), elements of the compute that used to be enclosed in quotations marks, such as strings, are now enclosed by apostrophes ('). For example, in the following compute the value 10 is enclosed in apostrophes:

```
<value compute="PAGE1.amountField.value + '10'"></value>
```

1.1.5 Namespaces in If/Then/Else Computes

If you are using an if/then/else compute that contains a namespace in the "then" portion, you must enclose the "then" portion in parentheses, as follows:

```
x ? (y) : z
```

For example:

```
FIELD1.value == '10' ? (FIELD1.custom:myOption) : FIELD2.value
```

If you do not enclose the "then" portion in parentheses, the parser will mistake the colon following the namespace prefix as the end of the "then" portion, and the compute will not function properly.

1.1.6 Multiline Computes

To place computes on multiple lines for easier readability, you must place a new line entity reference wherever you want to break a line. The new line entity reference is `
`

For example:

```
<custom:my_opt xfdl:compute = "toggle(activated, &#xA;
'off', 'on') == '1' ? &#xA;
set('convertedDate.value', &#xA;
sample_package.convertDate(theDate.value, &#xA;
```

theLocale.value)) : ""></custom:my_opt>

1.1.7 Supported Characters for Sid and Element Names

XFDL 6.0 supports the following Unicode characters in sid and element names (hex notation in parentheses):

- 0-9 (30 to 39)
- A-Z (41 to 5A)
- a-z (61 to 7A)
- _ (5F)
- (B7)
- (C0 to D6)
- (D8 to F6)
- (F8 to FF)

Sid and element names cannot start with:

- 0-9 (30 to 39)
- (B7)

Note that the \$ character is no longer allowed, and that the _ (underscore) character is no longer mandatory in custom item/option names.

1.1.8 Escaped Characters

You must escape the ampersand (&), apostrophe ('), and less than sign (<) to use these characters in computes or as character data.

To escape these characters, use the following encoding:

- & - &
- ' - '
- < - <

1.1.9 New Reserved Words ('or' and 'and')

'and' and 'or' have been added to the XFDL language as reserved words. They represent logical AND (&&) and logical OR (||), respectively.

1.1.10 Arrays

Options containing arrays no longer need to include the content="array" attribute within the option tag. For example, you would now write an itemlocation option as:

```
<itemlocation>
<ae>
  <ae>absolute</ae>
  <ae>276</ae>
  <ae>66</ae>
</ae>
<ae>
  <ae>extent</ae>
```

```

        <ae>246</ae>
        <ae>56</ae>
    </ae>
</itemlocation>

```

1.1.11 Additional Character Set Support

XFDL now supports multiple character sets for fonts, as specified by the `fontinfo` option. For more information, refer to the XFDL 6.0 Specification.

1.1.12 MIME Type For Forms

The `application/x-xfdl` MIME type is deprecated. All forms and applications should use the `application/vnd.xfdl` MIME type instead.

Support for the `application/x-xfdl` MIME type may be dropped in future versions.

1.1.13 Explicit Global Page and Global Items

XFDL now declares explicit global page and global items. The global page is marked with the `"globalpage"` tag, and global items are marked with the `"global"` tag. Both global pages and global items must have a `sid` of `"global"`. For example:

```

<?xml version="1.0"?>
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
      xmlns:xfdl="http://www.PureEdge.com/XFDL/6.0"
      <globalpage sid="global">
          <global sid="global">
              ...global form settings...
          </global>
      </globalpage>
      <page sid="Page1">
          <global sid="global">
              ...global page settings...
          </global>
          ...page contents...
      </page>
</XFDL>

```

For more information, refer to the XFDL 6.0 Specification.

1.1.14 New XFDL Function Calls

XFDL now supports the following function calls:

- `countDatagroupItems` - Allows you to count the total number of items in a particular datagroup.
- `countGroupedItems` - Allows you to count the total number of items in a particular group.
- `getAttr` - Allows you to get the value of an attribute on any element in a form.

- getPosition - Allows you to get the position index of an element within its parent.
- getPref - Allows you to get the value of any setting in Viewer Preferences form.
- getGroupedItem - Allows you to get the SID (scope identifier) of an item in a particular group.
- setAttr - Allows you to set the value of an attribute on any element in a form.
- xmlmodelUpdate - Allows you to update the XML Data Model in memory. This is useful if you have made computational changes to the data model.

For more information about these functions, refer to the XFDL Specification.

1.1.15 Change to Default Type for Actions and Buttons

Action and button items now default to a type of select.

1.1.16 New Encoding Attribute and Encoding Method

Options and array elements now support an encoding attribute that defines the encoding method used for the tag's contents. For example:

```
<mimedata encoding="base64"></mimedata>
```

Valid encoding methods are:

- xml
- base64
- base64-gzip

Note that base64-gzip is a new encoding method that gzips the contents and then converts them to base64.

The default encoding is xml, and all MIME data is stored in base64-gzip format.

Refer to the XFDL Specification for more information.

1.1.17 Support for Rich Text Fields

XFDL now supports rich text fields. These are enabled through the following new options:

- rtf
- texttype

Refer to the XFDL Specification for more information.

1.1.18 Default Display Value for Popup Items

If a popup does not have a label or a value option, it will be blank. Formerly, the popup displayed the value of the first cell.

1.1.19 Parent Operator for References

XFDL now supports a double dot (..) parent operator for references. The parent operator must appear at the beginning of a reference, and can appear multiple times to indicate additional ancestors. For example, the following are now valid references:

..FIELD1.value
...value

The first reference gets the parent of the current node, then locates the Field1 node, then the value node. The second gets the parent of the current node, then the parent of that node, then locates the value node.

Refer to the XFDL Specification for more information.

1.1.20 New Sign and Transmit Filters for Namespaces

XFDL supports two new filtering options: `signnamespaces` and `transmitnamespaces`. These options allow you to filter which namespaces are included in signatures and transmissions.

Refer to the XFDL specification for more information.

1.2 Viewer Features

1.2.1 Support for XFDL 6.0

The Viewer now supports XFDL 6.0 forms.

1.2.2 Support for Netscape 7

The Viewer now supports Netscape 7.

1.2.3 Dropped Support for UFDL

ICS Viewer no longer supports the Universal Forms Description Language (UFDL). Furthermore, files with a `.ufd` or `.ufdl` extension are no longer recognized and the `application/vnd.ufdl` MIME type is no longer supported.

1.2.4 Deprecated `.frm` File Extension

The `.frm` file extension has been deprecated. All new forms should use the `.xfd` or `.xfdl` file extension.

Support for the `.frm` file extension may be dropped in future versions.

1.2.5 Backwards Compatibility

ICS Viewer 6.0 supports XFDL version 4.4 and later. If a form of a version earlier than 4.4 is opened, the Viewer will process it as a version 4.4 form, and will apply the version 4.4 rules to the form.

Furthermore, ICS Viewer 6.0 is compatible with extensions (`.ifx` files) compiled with the version 4.5 API or later. Extensions compiled with earlier versions of the API will not work with this version of the Viewer.

1.2.6 Serialization of URLs in the `url` Option

If you list multiple URLs in the `url` option, the URLs are now processed in the order listed (formerly the order was undetermined).

For example, if a button of type submit listed 2 URLs, the submission would process the first URL and then the second URL.

However, if an action occurs that replaces or closes the original form (for example, a replace or done action) no further URLs are processed.

For example, if a button of type done listed 2 URLs, the submission would process the first URL then close the original form, preventing the second URL from being processed.

1.2.7 Removed Open File Shortcut

The Viewer no longer supports the ALT-CTRL-F key sequence for opening new files.

1.2.8 Character Set Support

The Viewer now supports multiple character sets for fonts, as specified by the fontinfo option. For more information, refer to the XFDL 6.0 Specification.

1.2.9 Activating the Screen Reader

The screen reader is no longer activated through a selection in the Viewer Preferences. Instead, the Viewer always makes screen reader information available. If JAWS is active, it will automatically recognize and read this information.

1.2.10 MAPI Support On By Default

The Viewer now defaults to using the active MAPI client when sending email. If you do not have a MAPI client available, you can change this setting in the Viewer Preferences.

1.2.11 New Zoom Feature

The Viewer now supports the ability to zoom forms from 25% to 200% of normal size. This feature is controlled by buttons on the Viewer's toolbar. Refer to the ICS Viewer User's Manual for more information.

1.2.12 Sizing of Items with label Options

Items for which the label option is set, but that have no border on the label, are now two pixels taller. This additional height accounts for the space the border would use if it were there. This may affect the spacing of items on any form that you upgrade to version 6.0.

1.2.13 Updated Shortcut Keys

The following shortcut keys have been changed:

CTRL + I	No longer opens the about dialog. It now toggles italics in rich text
----------	---

	fields.
--	---------

The following shortcut keys have been added:

CTRL + B	Toggles bold in rich text fields.
CTRL + F	Opens the Font dialog, which allows you to adjust font properties in rich text fields.
CTRL + G	Opens the Paragraph dialog, which allows you to adjust indent and justification in rich text fields.
CTRL + U.	Toggles underline in rich text fields.
CTRL + R	Right justifies text in rich text fields.
CTRL + SHIFT + PLUS	Increases the zoom (magnification) of the form.
CTRL + MINUS	Decreases the zoom (magnification) of the form.
CTRL + SHIFT + ?	Opens the about dialog.

1.2.14 Updated toggle Function

The implementation of the toggle function has been updated for version 6.0 forms to resolve outstanding problems. This may affect some forms that rely on unusual behavior. Be sure to test all forms that rely on the toggle function if you upgrade them to version 6.0.

1.2.15 Changed Default Height for Extent

Previously, if you did not provide a height for an extent (for example, if you used a compute to calculate the height but did not provide a starting value) the height would default to equal the width.

The height will now default to zero if no initial value is provided. This may impact forms that have expanding fields. If you want to recreate the old behavior, simply provide an initial value for the height.

1.2.16 Added New ufv_settings

The Viewer now supports the following new ufv_settings:

- aboutboxtext - Allows you add information to the Viewer's about box. This is useful if you want to add contact information, such as a name, address, and phone number.
- helpcursor - Allows you to control the appearance of the pointer when the Viewer is in help mode.

For more information, refer to the Using the Viewer Settings document available on the documentation web site.

1.2.17 Copying IFX Extensions from Older Viewers

The Viewer no longer copies IFX extension files from older versions of the Viewer during installation. You must copy these files manually.

1.2.18 Buttons and Labels with Empty Images

If a button or label's image option points to a data item that is empty or does not exist, then the button or label will display its value option instead. Formerly, the button or label displayed nothing.

2.0 Designer 2.3.0 Modifications

2.1 XFDL Changes

The Designer implements all of the XFDL changes listed in this section.

2.1.1 Namespace Support

Namespace support has been added to the XFDL language for XFDL documents version 6.0 and higher. Note that:

- The XFDL namespace URL is: <http://www.PureEdge.com/XFDL/6.0>
- The XFDL namespace is the default namespace for the document.
- All non-XFDL elements must have their own namespace.
- You must declare the default namespace, xfdl namespace, and any custom namespaces at the beginning of your XFDL form, immediately following the `<?xml version="x.x" ?>` tag.

For example:

```
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
xmlns:xfdl="http://www.PureEdge.com/XFDL/6.0"
xmlns:custom="http://www.PureEdge.com/XFDL/Custom">
```

2.1.2 Defining a Form's Version

With the introduction of namespace support, a form's XFDL version is now defined by the first namespace declaration on the XFDL tag. For example, a version 6.0 form will include the following XFDL tag:

```
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
```

However, you can still reference a form's version as though it were a global option, as shown:

global.global.version

2.1.3 XML Data Model

XFDL now supports an XML Data Model. This model allows form designers to create separate blocks of arbitrary XML within an XFDL form that share data with form elements, such as fields. This is useful for integrating with applications that require data in a particular XML format, such as schema-compliant data. The XML Data Model is based on the XForms standard, as published by the W3C, but is not limited to XForms. Refer to the XFDL Specification for more information.

2.1.4 XFDL Computes

XFDL computes are now attributes of the option tag. As a result, computes are contained in the opening tag of the option, and the current value of the compute is stored as simple character data in the option. The compute itself is enclosed by quotes. For example:

```
<field sid="FIELD1">  
  <value compute="PAGE1.nameField.value"  
  >Jane E. Smith</value>  
</field>
```

Additionally, because computes are now enclosed in quotation marks ("), elements of the compute that used to be enclosed in quotations marks, such as strings, are now enclosed by apostrophes ('). For example, in the following compute the value 10 is enclosed in apostrophes:

```
<value compute="PAGE1.amountField.value + '10'"></value>
```

2.1.5 Namespaces in If/Then/Else Computes

If you are using an if/then/else compute that contains a namespace in the "then" portion, you must enclose the "then" portion in parentheses, as follows:

$x ? (y) : z$

For example:

```
FIELD1.value == '10' ? (FIELD1.custom:myOption) : FIELD2.value
```

If you do not enclose the "then" portion in parentheses, the parser will mistake the colon following the namespace prefix as the end of the "then" portion, and the compute will not function properly.

2.1.6 Multiline Computes

To place computes on multiple lines for easier readability, you must place a new line entity reference wherever you want to break a line. The new line entity reference is `
`

For example:

```
<custom:my_opt xfdl:compute = "toggle(activated, &#xA;
'off', 'on') == '1' ? &#xA;
set('convertedDate.value', &#xA;
sample_package.convertDate(theDate.value, &#xA;
theLocale.value)) : ""></custom:my_opt>
```

2.1.7 Supported Characters for Sid and Element Names

XFDL 6.0 supports the following Unicode characters in sid and element names (hex notation in parentheses):

- 0-9 (30 to 39)
- A-Z (41 to 5A)
- a-z (61 to 7A)
- _ (5F)
- (B7)
- (C0 to D6)
- (D8 to F6)
- (F8 to FF)

Sid and element names cannot start with:

- 0-9 (30 to 39)
- (B7)

Note that the \$ character is no longer allowed, and that the _ (underscore) character is no longer mandatory in custom item/option names.

2.1.8 Escaped Characters

You must escape the ampersand (&), apostrophe ('), and less than sign (<) to use these characters in computes or as character data.

To escape these characters, use the following encoding:

- & - &
- ' - '
- < - <

2.1.9 New Reserved Words ('or' and 'and')

'and' and 'or' have been added to the XFDL language as reserved words. They represent logical AND (&&) and logical OR (||), respectively.

2.1.10 Arrays

Options containing arrays no longer need to include the content="array" attribute within the option tag. For example, you would now write an itemlocation option as:

```
<itemlocation>
  <ae>
    <ae>absolute</ae>
    <ae>276</ae>
    <ae>66</ae>
  </ae>
  <ae>
    <ae>extent</ae>
    <ae>246</ae>
    <ae>56</ae>
  </ae>
</itemlocation>
```

2.1.11 Additional Character Set Support

XFDL now supports multiple character sets for fonts, as specified by the fontinfo option. For more information, refer to the XFDL 6.0 Specification.

2.1.12 MIME Type For Forms

The application/x-xfdl MIME type is deprecated. All forms and applications should use the application/vnd.xfdl MIME type instead.

Support for the application/x-xfdl MIME type may be dropped in future versions.

2.1.13 Explicit Global Page and Global Items

XFDL now declares explicit global page and global items. The global page is marked with the "globalpage" tag, and global items are marked with the "global" tag. Both global pages and global items must have a sid of "global". For example:

```
<?xml version="1.0"?>
<XFDL xmlns="http://www.PureEdge.com/XFDL/6.0"
  xmlns:xfdl="http://www.PureEdge.com/XFDL/6.0"
  <globalpage sid="global">
    <global sid="global">
      ...global form settings...
    </global>
```

```
</globalpage>
<page sid="Page1">
  <global sid="global">
    ...global page settings...
  </global>
  ...page contents...
</page>
</XFDL>
```

For more information, refer to the XFDL 6.0 Specification.

2.1.14 New XFDL Function Calls

XFDL now supports the following function calls:

- `countDatagroupItems` - Allows you to count the total number of items in a particular datagroup.
- `countGroupedItems` - Allows you to count the total number of items in a particular group.
- `getAttr` - Allows you to get the value of an attribute on any element in a form.
- `getPosition` - Allows you to get the position index of an element within its parent.
- `getPref` - Allows you to get the value of any setting in Viewer Preferences form.
- `getGroupedItem` - Allows you to get the SID (scope identifier) of an item in a particular group.
- `setAttr` - Allows you to set the value of an attribute on any element in a form.
- `xmlmodelUpdate` - Allows you to update the XML Data Model in memory. This is useful if you have made computational changes to the data model.

For more information about these functions, refer to the XFDL Specification.

2.1.15 Change to Default Type for Actions and Buttons

Action and button items now default to a type of select.

2.1.16 New Encoding Attribute and Encoding Method

Options and array elements now support an encoding attribute that defines the encoding method used for the tag's contents. For example:

<mimedata encoding="base64"></mimedata>

Valid encoding methods are:

- xml
- base64
- base64-gzip

Note that base64-gzip is a new encoding method that gzips the contents and then converts them to base64.

The default encoding is xml, and all MIME data is stored in base64-gzip format. Refer to the XFDL Specification for more information.

2.1.17 Support for Rich Text Fields

XFDL now supports rich text fields. These are enabled through the following new options:

- rtf
- texttype

Refer to the XFDL Specification for more information about the new options.

2.1.18 Default Display Value for Popup Items

If a popup does not have a label or a value option, it will be blank. Formerly, the popup displayed the value of the first cell.

2.1.19 Parent Operator for References

XFDL now supports a double dot (..) parent operator for references. The parent operator must appear at the beginning of a reference, and can appear multiple times to indicate additional ancestors. For example, the following are now valid references:

```
..FIELD1.value  
...value
```

The first reference gets the parent of the current node, then locates the Field1 node, then the value node. The second gets the parent of the current node, then the parent of that node, then locates the value node.

Refer to the XFDL Specification for more information.

2.1.20 New Sign and Transmit Filters for Namespaces

XFDL supports two new filtering options: signnamespaces and transmitnamespaces. These options allow you to filter which namespaces are included in signatures and transmissions.

Refer to the XFDL specification for more information.

2.2 Designer Features

2.2.1 Support for XFDL 6.0.0

The Designer now produces version 6.0.0 XFDL forms

2.2.2 Dropped Support for UFDL

ICS Designer no longer supports the Universal Forms Description Language (UFDL). Furthermore, files with a .ufd or .ufdl extension are no longer recognized and the application/vnd.ufdl MIME type is no longer supported.

2.2.3 Deprecated .frm File Extension

The .frm file extension has been deprecated. All new forms should use the .xfd or .xfdl file extension.

Support for the .frm file extension may be dropped in future versions.

2.2.4 Backwards Compatibility

ICS Designer 2.3 opens forms in any version of XFDL. However, the following limitations apply:

Forms prior to 4.4 - The Designer can read these forms, but will interpret them as version 4.4 forms and apply version 4.4 rules to them.

Forms prior to 5.1 - The Designer can read these forms, but cannot display them until they are upgraded to version 5.1 or later. When a form prior to version 5.1 is opened, the user is immediately prompted to upgrade the form.

Furthermore, ICS Designer 2.3 is compatible with extensions (.ifx files) compiled with the version 4.5 API or later. Extensions compiled with earlier versions of the API will not work with this version of the Designer.

2.2.5 Custom Item Creation

The Designer no longer supports the creation of custom items through the normal user interface. To create custom items, you must now use Code View.

2.2.6 Designer Retains Window Size

The Designer now remembers whether the form window should be maximized when opening a form. This setting is based on the window size the user last worked in. For

example, if the user maximizes the form window and then closes the Designer, the Designer will open the next form in a maximized window.

2.2.7 Designer Sets Fields to Wordwrap by Default

When you create Fields, the Designer now automatically sets them to wordwrap.

2.2.8 New Shortcut Keys

The Designer now supports the following shortcut keys:

- CTRL + F1 - Bring to front.
- CTRL + F2 - Send to back.

2.2.9 Designer Requires Administrator Access to Install

The Designer Install package no longer works under Power User access. You now need Administrator access to install the Designer under later versions of Windows.

Note that once the Designer is installed, you can still run it as either an Administrator or a Power User.

Software Installation Instructions

For Designer 2.3, the following instructions apply:

1. It is recommended that the previous version of IMT Designer is uninstalled prior to installing the newer version, unless there is a business requirement to maintain multiple installed versions. To uninstall, the standard Windows procedure is used (Add/Remove Program utility)
2. Locate the product InstallShield executable
 - ICS Designer 2.3 - ICSDesigner230.exe
3. Double-click the executable. This will launch the InstallShield application which will provide any necessary assistance to the person installing the software.

For Viewer 6.0, the following instructions apply:

1. It is recommended that the previous version of IMT Viewer is uninstalled prior to installing the newer version, unless there is a business requirement to maintain multiple installed versions. To uninstall, the standard Windows procedure is used (Add/Remove Program utility)
2. Locate the product InstallShield executable
 - ICS Viewer 6.0 – ICSViewer600DODJ12.exe

3. Double-click the executable. This will launch the InstallShield application which will provide any necessary assistance to the person installing the software.
4. Create the directory "DODJ12TrustPoints" in the required location and populate with the Air Force specified trusted root extracts (see document "Creating JITC certified forms.pdf" for more details)

In addition, the IMT Designer software has been augmented for AF use to include a tool that allows designers to specify metadata. Metadata is composed of approximately 20 fields of information describing the IMT such as IMT name, IMT date, prescribing directive, etc. Designers input metadata through a properties-like page that is made available when the metadata.exe file and associated config.xml (providing valid values) are loaded onto Designers' computers.

Metadata.exe (and associated custom Air Force config.xml file) are currently specifically for USAF use and are not bundled in the Designer 2.3 off-the-shelf release.

Both of these files will need to be distributed separately to all Designer workstations. The files must be copied per instruction below.

- Copy config.xml and metadata.exe to C:\Program Files\PureEdge\Designer 2.3\Modules
- Launch Designer
- With a form open (new or existing) the "Tools" menu should now have a "Metadata" option at the very bottom.

If a workstation is being upgraded from release 2.2 to release 2.3, these files will not need to be re-installed. When the administrator uninstalls v2.2, everything is removed except for custom components such as metadata.exe and config.xml. These components can then be transferred from C:\Program Files\PureEdge\Designer 2.2\Modules directory to the C:\Program Files\PureEdge\Designer 2.3\Modules directory location

Security or privacy information relevant to capability upgrade installation: Not applicable.

Section 2. Communications Paths

CMS Phase 1.1 utilizes current AF communications (e.g. NIPRNET and SIPRNET) with no change to existing networks.

Users can obtain IMTs from a variety of sources, which will vary depending on local policy. As depicted in Figure 2.1, the forms user can obtain designed IMTs from the e-publishing web site, from CDROM, and/or from local distribution. The e-publishing web site will contain the most current IMTs and allows users to quickly search for and download blank IMTs.

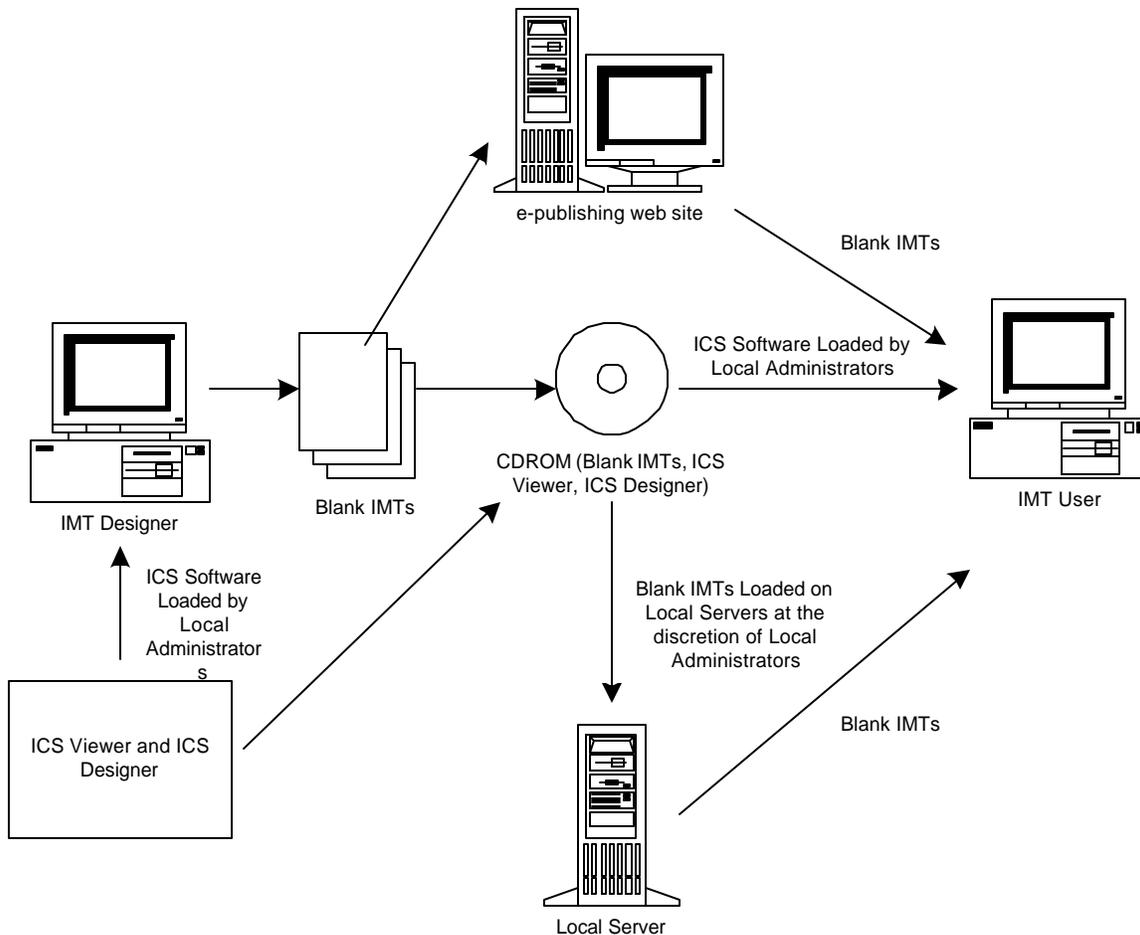


Figure 2.1: IMT Operational Node Connectivity (OV-2)

In summary, the following activities will occur:

- Designers will design blank IMTs, which will be made available to IMT users via the e-publishing web site, CDROM, and local distribution
- Users search and download blank IMTs from above distribution sources
- Users fill and edit electronic IMTs
- Users save filled or partially filled IMTs to local directories
- Users print filled IMTs
- Users route filled IMTs via existing email systems to desired recipients

Figure 2.2 depicts a detailed breakdown of IMT handling processes as they will occur under the SIPRNET:

- IMT Software (IMT Viewer and IMT Designer) loaded on user PC's by local administrators
- Blank IMTs are available to SIPRNET users via CDROM and diskette
- SIPRNET users fill and edit electronic IMTs

- SIPRNET users save filled or partially filled IMTs to local directories and route to other SIPRNET users

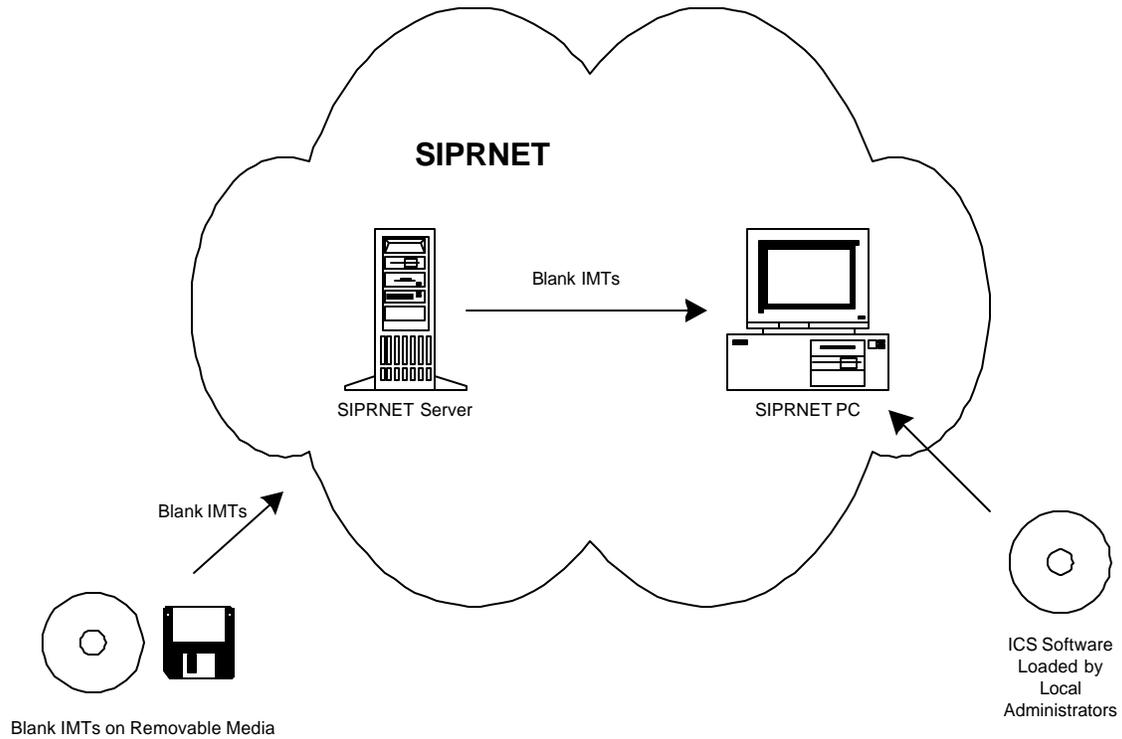


Figure 2.2: IMT SIPRNET Processes

Section 3. Per AFCA guidance, an update SSAA is not required pending AFCA review of this Network Determination Checklist.

Networthiness Determination Questionnaire

Section 4. Server Concerns	Yes	No	N/A
4a. Does the IT capability reside on a server?	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
4b. If Yes , Does the server reside within the boundaries of the Air Force Enterprise network?	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4c. If Yes , An AF Certificate of Networthiness is required. Continue with the Certificate of Networthiness Determination Checklist (Section 5 or 6) below. If No , Does the user connect via a web browser interface?	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4d. If no , an AF Certificate of Networthiness is required. Continue with the Certificate of Networthiness Determination Checklist (section 5 or 6) below.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
4e. If yes , does the server download mobile code to the browser, i.e. Active X, Javascript, etc. or require a plug-in program for the browser?	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
If yes , An AF Certificate of Networthiness is required. Continue with the Certificate of Networthiness Determination Checklist below. If no , the IT capability may require an AF Certificate of Networthiness but requires more detailed information to determine. Contact AFCA/WFC or your already appointed AFCA/WF Action Officer for further guidance.	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

NOTE: For new IT capabilities and operational capabilities not having a Certificate of Networthiness, begin at Section 5. For modifications and upgrades of IT capabilities previously issued a CoN, begin at Section 6.

Per above guidance, please see Section 6.

<h2 style="text-align: center;">New IT Capabilities</h2> <h3 style="text-align: center;">Section 5. CONNECTIVITY ISSUES</h3>	Yes	No	N/A
<p>5(a). Are security products or mechanisms used to protect the IT capability? If Yes, identify and explain. Examples include:</p> <ul style="list-style-type: none"> • Encryption tunnels. • Intrusion detection systems (IDS). • Implementation of localized (functional community boundary) security (such as a firewall). • Passwords. • Password auditing. • Public Key Encryption • Other _____ 			
<p>5(b). Does the IT capability require changes to standard firewall ports or protocols/network services?</p>			
<p>5(c). Is the IT capability web-based? If yes please explain: _____ _____ _____</p>			
<p>Note: For all explanations of more than three lines, provide narrative as an attachment. Reference the attachment in the corresponding questionnaire block.</p>			
<p>5(d). Is the IT capability being added to the AF Portal or making a change to the AF portal or AF portal firewall? If yes please explain: _____ _____ _____</p>			
<p>5(e). Are there specific requirements for external interfaces/connections/ additional ports/protocols (e.g., dial-up capability, connections to a non .mil site, remote administration/ management)? If yes please explain: _____ _____ _____</p>			
<p>5(f). Are there any wireless transmission capabilities included (e.g., radio frequency, infrared, Point to Point, Point to Multi-Point, laser WLAN, etc)? If yes please explain: _____ _____ _____</p> <p>Note: All wireless procurements must be IAW DoD and MILDEP directives. SME AFFMA DSN 328-1535.</p>			

IDENTIFICATION AND AUTHENTICATION (Includes Auditing)			
5(g). Have password generation requirements or password management been implemented? Note: Reference AFMAN 33-223 If yes , describe these password implementations in terms of: <ul style="list-style-type: none"> • Support to AF/DoD password policy • Automated means to enforcing password policy • Enforcement of password changes every 90 days • Lockout features after unsuccessful login attempts. 			

Modified / Upgraded IT Capabilities	Yes	No	N/A
Section 6. CONNECTIVITY ISSUES			
6(a). For upgrades/modifications: Have any changes been made to security mechanisms? Include para references in SSAA or DITSCAP direction. If yes please explain: No changes have been made to /API 6.0 or Designer 2.3 related to security		x	
6(b). Does the modification/upgrade require changes to standard firewall ports or protocols/network services?		x	
6(c). Is this IT capability version web-based? If yes please explain: _____ _____ _____			
6(d). Is the capability being added to the AF Portal or making a change to the AF portal or AF portal firewall? If yes please explain: _____ _____ _____			
6(e). Are there required changes to any external interfaces/connections/ additional ports/protocols (e.g., dial-up capability, connections to a non .mil site, remote administration/ management)? If yes please explain: _____ _____ _____			

<p>6(f). Has user/administrator access changed (e.g., added Foreign National access, or expanded off .mil domain)? If yes please explain: _____ _____ _____</p>			
<p>6(g). Are there any wireless transmission capabilities now included or changed (e.g., radio frequency, infrared, Point to Point, Point to Multi-Point, laser WLAN, etc)? If yes please explain: _____ _____ _____</p> <p>Note: All wireless procurements must be IAW DoD and MILDEP directives. SME AFFMA DSN 328-1535.</p>			
<p>6(h). Has the using community changed or has the using community changed their location (e.g., users now reside outside the base security perimeter (either individual users via RAS or tenant organizations that are not behind the base security perimeter, such as Air National Guard or Joint offices)? If yes please explain: _____ _____ _____</p>			
<p>6(i). Have the modifications changed network bandwidth requirements? If yes, by what magnitude? _____ (i.e., throughput increased/decreased by X amount)</p>			
<p>6(j). Has supporting application software changed (e.g., Oracle 8i to MS SQL Server)? If yes please explain: _____ _____ _____</p>			
<p>6(k). Have supporting architectures changed. (e.g., changing server locations or topology, local server to centralized server)? If yes please explain: _____ _____ _____</p>			
<p>IDENTIFICATION AND AUTHENTICATION (Includes Auditing)</p>			
<p>6(l). Have any changes to user account management procedures or processes been implemented. (i.e. changed definition of administrator duties, location, privileges, super user, etc.) If yes please explain: _____ _____ _____</p>			

<p>6(m). Have any changes to password generation requirements or password management been implemented? Note: Reference AFMAN 33-223 If yes, describe the changes to password generation/management in terms of:</p> <ul style="list-style-type: none"> • Support to AF/DoD password policy • Automated means to enforcing password policy • Enforcement of password changes every 90 days • Lockout features after unsuccessful login attempts. <hr/> <hr/> <hr/>			
ROLES AND RESPONSIBILITIES			
<p>6(n). Have any changes in support postures (roles or responsibilities) been introduced? If yes please explain: _____</p> <hr/> <hr/>			
MISCELLANEOUS			
<p>6(o). Has the Operating System or supporting COTS products changed in any way? (e.g., database, encryption software, user interface...) If yes please explain: _____</p> <hr/> <hr/>			
<p>6(p). Have the data storage requirements changed? (e.g., magnitude of size, location, backup capabilities, etc.) If yes please explain: _____</p> <hr/> <hr/>			
<p>6(q). Has data compilation or aggregation of data changed? (classification of data, classification of aggregation of data...) If yes please explain: _____</p> <hr/> <hr/>			