



Grandstream Networks, Inc.

基于 XML 的屏保定制指南

GXV21XX IP 电话/GXP1450 IP 电话

目录

概述	3
IT原理及构造	3
XML语法	3
自定义屏保举例	9
自定义Logo	9
自定义软按键	9
自定义账号状态	10
附录	10
基本要素 “Screen”	10
要素“LeftStatusBar”	11
要素“IdleScreen”	11
要素 “DISPLAYSTR”	11
GXP21XX/GXP1450 屏幕/Logo大小	12

概述

GXP21xx 和 GXP1450 支持基于 XML 的自定义屏保。一些客户提出需要一个用于定制界面显示的 API 接口后，自定义屏保的显示信息和布局设计都高度依赖于用户的喜好和需求。

本文详细介绍了潮流网络 XML 可定制界面 API 接口的设计，此接口可以用于定制 GXP21xx 和 GXP1450 的屏保。

IT 原理及构造

新引入的配置项设置如下：

- 是否允许下载自定义屏保 (P340)：是/否，是/HTTP，TFTP（默认为否）。
可选值：0（否）/1（HTTP）/2（TFTP），忽略其他值。
- 重启时自动下载屏保 (P1349)：是/否（默认为否）
可选值：0（否）/1（是）忽略其他值。
- 使用个别文件名(P1343)：是/否（默认为否）
可选值：0（否）/1（是）忽略其他值。
- 屏保下载服务器路径(P341)：包含 XML 文件路径的字符串，最大 128 字符。必须是主机/路径格式，例如：“directory.grandstream.com/engineering”。

当用户将“是否允许下载自定义屏保”参数设置为“是”，并且将参数“屏保下载服务器路径”设置为正确的路径，就会激活下载自定义屏保功能。

此功能在刚激活时不会马上下载自定义屏保 XML 文件。用户可以将参数“重启时自动下载屏保”设置为“是”，话机在下次重启时将自动下载此 XML 文件。

在话机屏幕菜单中的个性化设置中，新增了如下两个选项：

- 下载屏幕保护
- 清除自定义屏保

用户可以通过话机菜单中选择“个性化设置”>“下载屏幕保护”选项启动屏保下载进程。话机将会到参数“屏保下载服务器路径”指定的路径中下载 XML 文件。

如果参数“使用个别文件名”设置为“否”，则 XML 文件必须命名为 gs_screen.xml。参数“屏保下载服务器路径”设置为存放 gs_screen.xml 文件的文件夹路径，例如：
grandstream.com/directory/custfiles。

如果参数“使用个别文件名”设置为“是”，则用户可以根据自己的喜好为 XML 文件命名。在这种情况下，参数“屏保下载服务器路径”必须指定具体的文件名，例如：
grandstream.com/directory/custfiles/gxp2100xmlcust.xml。

当话机成功下载 XML 屏保文件后，会立即使其生效。此文件也会在本机保存备用。

XML 语法

用户可以通过 XSD 文件大致了解 XML 文件的样式。用户不需要手动检查文件的语法结构，这一过程可以自动化。可以实现这一自动化过程的网站：<http://www.xmlme.com/Validator.aspx>。

本节将详细介绍如何使用上文提及的网站自动检查 XML 文件的语法结构。

1. 将网站上的 XML 结构区域和 XML 文档区域清空。

2. 复制下文所示的 XSD 文件到 XML 结构区域。
3. 将用户屏保文件复制到 XML 文档区域。
4. 单击“validate”。
5. 查看验证结果是否成功。如果验证失败，证明此 XML 文件存在结构错误。

XSD 文件:

```
<?xml version="1.0"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Screen">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="LeftStatusBar" minOccurs="1" maxOccurs="1" >
          <xs:complexType>
            <xs:sequence>
              <xs:element name="Layout" type="GeneralLayoutTemplateType" minOccurs="1"
maxOccurs="1"></xs:element>
              <xs:element name="Account" type="GeneralTemplateType" minOccurs="1" maxOccurs="1"></xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="IdleScreen" minOccurs="1" maxOccurs="4">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="ShowStatusLine" type="xs:boolean" minOccurs="0" maxOccurs="1" default="true"/>
              <xs:element name="DisplayBitmap" type="BitmapType" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
              <xs:element name="DisplayString" type="StringType" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
              <xs:element name="DisplaySet" type="GeneralListHolderType" minOccurs="0" maxOccurs="1"
nillable="true"></xs:element>
              <xs:element name="SoftKeys" minOccurs="0" maxOccurs="1" nillable="true">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="Layout" type="GeneralLayoutTemplateType" minOccurs="1"
maxOccurs="1"></xs:element>
                    <xs:element name="ButtonShape" type="GeneralTemplateType" minOccurs="1"
maxOccurs="unbounded"></xs:element>
                    <xs:element maxOccurs="12" name="SoftKey" type="SoftKeyType">
                      </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

<xs:complexType name="GeneralLayoutTemplateType">
  <xs:sequence>
    <xs:element name="DisplayList" type="ListHolderType" minOccurs="1" maxOccurs="1"/>
    <xs:element name="DisplayBitmap" type="BitmapType" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
    <xs:element name="DisplayString" type="StringType" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
  <xs:attribute name="height" type="xs:int" use="optional" default="0"/>
  <xs:attribute name="width" type="xs:int" use="optional" default="0"/>
  <xs:attribute name="padding-top" type="xs:int" use="optional" default="0"/>
  <xs:attribute name="padding-bottom" type="xs:int" use="optional" default="0"/>
</xs:complexType>

<xs:complexType name="GeneralTemplateType">
  <xs:sequence>
    <xs:element name="DisplayBitmap" type="BitmapType" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
    <xs:element name="DisplayString" type="StringType" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
  </xs:sequence>
  <xs:attribute name="height" type="xs:integer"/>
  <xs:attribute name="width" type="xs:integer"/>
  <xs:attribute name="id" type="xs:ID" use="optional"/>
</xs:complexType>

<xs:complexType name="ListHolderType">
  <xs:sequence>
    <xs:element name="X" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
    <xs:element name="Y" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
  </xs:sequence>
</xs:complexType>

<!-- DataList type -->
<xs:complexType name="GeneralListHolderType">
  <xs:sequence>
    <xs:element name="ItemTemplate" type="GeneralTemplateType" minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"/>
  <xs:attribute name="x" type="xs:integer" default="0"/>
  <xs:attribute name="y" type="xs:integer" default="0"/>
  <xs:attribute name="maxNumberOfRecord" type="xs:integer" default="1"/>

```

```
<xs:attribute name="dataSourceName" type="xs:string" use="required"/><!-- [$Stock, $Currency, ($NewsFeed),
($Weather) and so on] -->
<xs:attribute name="displayDirection" default="vertical">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="vertical"/>
      <xs:enumeration value="horizontal"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
</xs:complexType>

<xs:complexType name="BitmapType">
  <xs:sequence>
    <xs:element name="Bitmap" type="xs:string" minOccurs="1" maxOccurs="1"/>
    <xs:element name="X" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
    <xs:element name="Y" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
    <xs:element name="displayCondition" type="IdleScreenConditionType" minOccurs="0" maxOccurs="1"
nillable="true"/>
  </xs:sequence>
  <xs:attribute name="isfile" type="xs:boolean"/>
  <xs:attribute name="isflash" type="xs:boolean"/>
  <xs:attribute name="isrenew" type="xs:boolean"/>
</xs:complexType>

<xs:complexType name="StringType">
  <xs:sequence>
    <xs:element name="DisplayStr" type="xs:string" minOccurs="1" maxOccurs="1" />
    <xs:element name="X" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
    <xs:element name="Y" type="xs:integer" minOccurs="1" maxOccurs="1" default="0"/>
    <xs:element name="displayCondition" type="IdleScreenConditionType" minOccurs="0" maxOccurs="1"
nillable="true"/>
  </xs:sequence>
  <xs:attribute name="font" default="regular">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="regular"/>
        <xs:enumeration value="bold"/>
        <xs:enumeration value="unifont"/>
        <xs:enumeration value="time"/>
        <xs:enumeration value="numberfont"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
```

```
</xs:attribute>
<xs:attribute name="halign" default="left">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="left"/>
      <xs:enumeration value="center"/>
      <xs:enumeration value="right"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="color" default="Black">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Black"/>
      <xs:enumeration value="White"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="isflash" type="xs:boolean"/>
</xs:complexType>

<xs:complexType name="IdleScreenConditionType">
  <xs:sequence>
    <xs:element name="conditionType">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="alwaysDisplay"/>
          <xs:enumeration value="missCall"/>
          <xs:enumeration value="callFwded"/>
          <xs:enumeration value="callFwdCancelled"/>
          <xs:enumeration value="dnd"/>
          <xs:enumeration value="headsetMode"/>
          <xs:enumeration value="signIn"/>
          <xs:enumeration value="signOut"/>
          <xs:enumeration value="backSpace"/>
          <xs:enumeration value="XmlApp"/>
          <xs:enumeration value="SubScreen"/>
          <xs:enumeration value="nfsMountOk"/>
          <xs:enumeration value="nfsMountFailed"/>
          <xs:enumeration value="coredump"/>
          <xs:enumeration value="networkUp"/>
          <xs:enumeration value="hasIM"/>
          <xs:enumeration value="hasVM_IM"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```

```
    <xs:enumeration value="hasVoiceMail"/>
    <xs:enumeration value="hasDialedCallog"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
<xs:attribute name="negate" type="xs:boolean" default="false"/>
</xs:complexType>
```

```
<xs:complexType name="EmptyType">
  <xs:sequence />
</xs:complexType>
```

```
<xs:complexType name="IdleActionType">
  <xs:choice>
    <xs:element name="CallReturn" type="EmptyType" />
    <xs:element name="NewCall" type="EmptyType" />
    <xs:element name="FwdAll" type="EmptyType" />
    <xs:element name="CancelFwd" type="EmptyType" />
    <xs:element name="SignIn" type="EmptyType" />
    <xs:element name="SignOut" type="EmptyType" />
    <xs:element name="BackSpace" type="EmptyType" />
    <xs:element name="CANCEL" type="EmptyType" />
    <xs:element name="Redial" type="EmptyType" />
    <xs:element name="MissedCalls" type="EmptyType" />
    <xs:element name="SwitchSCR" type="EmptyType" />
    <xs:element name="ReverseCurrency" type="EmptyType" />
  </xs:choice>
</xs:complexType>
```

```
<xs:complexType name="SoftKeyType">
  <xs:sequence>
    <xs:element name="Label" minOccurs="0" maxOccurs="1">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="1"/>
          <xs:maxLength value="32"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="Icon" minOccurs="1" maxOccurs="1" type="xs:string">
      <xs:attribute name="x" type="xs:integer"/>
    </xs:element>
  </xs:sequence>
</xs:complexType>
```



```
<xs:attribute name="y" type="xs:integer"/>
<xs:attribute name="textoffset" type="xs:integer"/>
<xs:attribute name="isfile" type="xs:boolean"/>
</xs:element>
<xs:element name="Action" minOccurs="1" maxOccurs="1" type="IdleActionType">
</xs:element>
<xs:element name="displayCondition" type="IdleScreenConditionType" minOccurs="0" maxOccurs="1"
nillable="true"/>
</xs:sequence>
<xs:attribute name="useshapeid" type="xs:string" use="optional"/>
</xs:complexType>

</xs:schema>
```

自定义屏保举例

在 XML 自定义屏保文件包中，已经提供了用于 GXP2100、GXP2110、GXP2120 和 GXP1450 话机的默认的自定义屏保 XML 文件。用户只需要在默认的模板中做少量修改，就可以自定义 Logo、软按键等屏幕显示项。完成修改后，如果用户不使用个别文件名，则需要将 XML 文件命名为 `gs_screen.xml`。

自定义 Logo

默认的屏保 Logo 是 Grandstream(潮流网络公司 Logo)。用户可以根据如下所示的步骤修改文件包中的默认 XML 文件，自定义屏保 Logo。

1. 在文件包中找到与您的话机型号相匹配的默认 XML 文件。
2. 针对不同的屏幕显示，在 XML 文件中共有四个不同内容的“IdleScreen”小节。每一小节的设置都可以通过按话机上的“切换”软按键显示在屏幕上。
3. 找到第一个“IdleScreen”小节，对应于话机的第一个默认屏保。
4. 将 `<DisplayBitmap isfile="true">` 修改为 `<DisplayBitmap isfile="false">`。
5. 用户首先需要编辑图片，然后将编辑好的图片插入到 `<Bitmap>` 标记对中的相应位置，即 `"/app/resource/logo/g_s_logo.bmp"`处，如下文所示：

```
<DisplayBitmap isfile="false">
    <Bitmap>/app/resource/logo/g_s_logo.bmp</Bitmap>
    <X>15</X>
    <Y>26</Y>
</DisplayBitmap>
```

6. 在 Web 界面中的高级设置页面，通过 TFTP/HTTP 方式上传编辑好的自定义屏保文件，然后点击“更新”并重启设备。
7. Logo 图片上传后，账号设置信息仍旧显示在屏幕的左边，如果用户不希望显示左边的账号信息窗口，也可以在此文件中自定义。
8. 用户可以通过修改 X、Y 的值来调整 Logo 在屏幕上的显示位置。

自定义软按键

在 xml 文件中有四个定义屏保页面的小结，每一个小结中都包含不一样的软按键。用户可以通过修改每个 `<IdleScreen>` 小节中的 `<Softkeys></Softkeys>` 标记对的内容来编辑软按键。例如，在屏幕上隐藏

“重拨”软按键的步骤如下：

1. 在文件包中找到与您的话机型号相匹配的默认 XML 文件。
2. 针对不同的屏幕显示，在 XML 文件中共有四个不同内容的“IdleScreen”小节。每一小节的设置都可以通过按话机上的“切换”软按键显示在屏幕上。
3. 找到第一个“IdleScreen”小节，对应于话机的第一个默认屏保。
4. 在“IdleScreen”小节中找到“Softkeys”小节，将重拨软按键的内容删除，如下文所示：

```
<SoftKey>
  <Action>
    <Redial/>
  </Action>
  <displayCondition>
    <conditionType>hasDialedCalllog</conditionType>
  </displayCondition>
</SoftKey>
```

5. 在 Web 界面中的高级设置页面，通过 TFTP/HTTP 方式上传编辑好的自定义屏保文件，然后点击“更新”并重启设备。
6. “重拨”软按键隐藏后，终端用户将不能通过软按键重拨号码。

自定义账号状态

话机的每个屏保页面的左侧都会默认显示账号状态。用户可以根据自己的喜好隐藏账号状态。在显示天气信息的屏保页面隐藏账号状态的步骤如下：

1. 在文件包中找到与您的话机型号相匹配的默认 XML 文件。
2. 针对不同的屏幕显示，在 XML 文件中共有四个不同内容的“IdleScreen”小节。每一小节的设置都可以通过按话机上的“切换”软按键显示在屏幕上。
3. 找到与显示天气信息的页面相对应的“IdleScreen”小节。
4. 找到标记对<ShowStatusLine>true<ShowStatusLine>。
5. 将“true”修改为“false”。
6. 在 Web 界面中的高级设置页面，通过 TFTP/HTTP 方式上传编辑好的自定义屏保文件，然后点击“更新”并重启设备。
7. 上传完成后，用户可以在话机上按“切换”软按键切换到显示天气信息的页面。此时账号信息被隐藏，界面上的其它内容自动调整显示位置以适应新的屏保画面。

附录

基本要素“Screen”

XML 文档中有一个基本要素“Screen”，它包含两个必要的子要素“LeftStatusBar”和“IdleScreen”。

```
<xs:element name="Screen">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="LeftStatusBar" minOccurs="1" maxOccurs="1" />
    </xs:element>
    <xs:element name="IdleScreen" minOccurs="1" maxOccurs="4"/>
  </xs:element>
</xs:sequence>
</xs:complexType>
```

```
</xs:element>
```

要素 “LeftStatusBar”

此要素包含两个必要的子要素 “Layout” 和 “Account”，这两个元素控制通用模板的布局和账号状态。

```
<xs:element name="LeftStatusBar" minOccurs="1" maxOccurs="1" >
  <xs:complexType>
    <xs:sequence>
      <xs:element name="Layout" type="GeneralLayoutTemplateType" minOccurs="1" maxOccurs="1"></xs:element>
      <xs:element name="Account" type="GeneralTemplateType" minOccurs="1" maxOccurs="1"></xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

要素 “IdleScreen”

此要素定义了屏保的 5 个组成部分，这些组成部分也以要素的形式定义。

```
<xs:element name="IdleScreen" minOccurs="1" maxOccurs="4">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="ShowStatusLine" type="xs:boolean" minOccurs="0" maxOccurs="1" default="true"/>
      <xs:element name="DisplayBitmap" type="BitmapType" minOccurs="0" maxOccurs="unbounded"
nillable="true"/>
      <xs:element name="DisplayString" type="StringType" minOccurs="0" maxOccurs="unbounded" nillable="true"/>
      <xs:element name="DisplaySet" type="GeneralListHolderType" minOccurs="0" maxOccurs="1" nillable="true"/>
      <xs:element name="SoftKeys" minOccurs="0" maxOccurs="1" nillable="true"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

说明：在上述语法中 “ShowStatusLine”、“DisplaySet” 和 “SoftKeys” 可以禁用或启用（仅出现一次）， “DisplayBitmap” 和 “DisplayString” 可以有任意数量的实例。

显示规则：当 “DisplayBitmap” 和 “DisplayString” 同时出现时，所有的位图都将比字符串先显示。当多个(bitmap/string)类型同时出现，它们在屏幕上的显示顺序和它们在 XML 文档中的顺序相同，并且后显示的内容会覆盖先显示的内容。

要素 “DISPLAYSTR”

此要素包含了需要显示的字符串。

```
<xs:element name="DisplayStr" type="xs:string" minOccurs="1" maxOccurs="1" />
```

此字符串包含了动态内容。在话机运行时，将会被动态内容替代的重要系统变量如以下列表所示。

1. **\$W**：此变量将会被替换为一周中的当前日期。有效取值为：周日、周一、周二、周三、周四、周五、周六。
2. **\$N**：此变量将会被替换为实际配置的账号 1 显示名称。
3. **\$X**：此变量将会被替换为实际配置的账号 1 用户号码。

4. **\$V**: 此变量将会被替换为实际配置的 SIP 服务器。
5. **\$I**: 此变量将会被替换为系统 IP 地址。
6. **\$D**: 此变量将会被替换为一个月中的当前日期, 其中个位数的日期前面加 0。有效取值为: 01、02、03、...、31。
7. **\$d**: 此变量将会被替换为一个月中的当前日期, 其中个位数的日期前面不加 0。有效取值为: 1、2、3、...、31。
8. **\$M**: 此变量将会被替换为英文的月份名称。有效取值为: January, February, ..., December。
9. **\$o**: 此变量将会被替换为月份的数值, 其中个位数的月份前加 0。有效取值为: 01, 02, ..., 12。
10. **\$n**: 此变量将会被替换为月份的数值, 其中个位数的月份前不加 0。有效取值为: 1, 2, ..., 12。
11. **\$Y**: 此变量将会被替换为四位数的年份, 例如: 2006, 2007
12. **\$y**: 此变量将会被替换为两位数的年份, 例如: 06, 07 ...
13. **\$P**: 此变量将会被替换为大写的 AM/PM 状态。有效取值为: AM, PM。
14. **\$p**: 此变量将会被替换为小写的 am/pm 状态。有效取值为: am, pm。
15. **\$h**: 此变量将会被替换为当前的小时数, 以 12 小时制显示, 个位数前加 0。有效取值为: 01, 02, ..., 12。
16. **\$m**: 此变量将会被替换为当前的分钟数, 个位数前面加 0。有效取值为: 01, 02, ..., 59
17. **\$s**: 此变量将会被替换为当前的秒数, 个位数前面加 0。有效取值为: 01, 02, ..., 59

说明:

- 1、如果用户不需要显示“\$”符号, 请使用“\$\$”转义字符。
- 2、随着固件版本的升级将会增加新的变量值。

GXP21XX/GXP1450 屏幕/Logo 大小

型号	屏幕大小 (像素)	默认 Logo 大小 (像素)	默认 Logo 区域大小 (像素)
GXP2100	180 x 90	72 x 35	123 x 58
GXP2120	320 x 160	136 x 66	230 x 85
GXP2110	240 x 120	89 x 46	165 x 50
GXP1450	180 x 60	N/A	N/A