Chromium Resource

目录

1 GRD 文件

Chromium 的资源包括字符串、和文件等；Chromium 以 grd 文件组织这些资源。顾名思义，grd 是 Google Resource Define 的意思。一个 grd 文件是一个 xml 文件，典型的 grd 文件示例如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<grit latest_public_release="0" current_release="1">
  <outputs>
    <output filename="grit/locale_settings.h" type="rc_header">
      <emit emit_type='prepend'></emit>
    </output>
    <output filename="locale_settings_en-US.pak"
           type="data_package" lang="en" />
    <output filename="platform_locale_settings_zh-CN.pak"
           type="data_package" lang="zh-CN" />
  </outputs>
  <translations>
    <file path="platform_locale_settings/locale_settings_win_zh-CN.xtb"
        lang="zh-CN" />
  </translations>
  <release>
    <messages fallback_to_english="true">
      <message name="IDS_FIXED_FONT_FAMILY" use_name_for_id="true">
            Courier New
      </message>
    </messages>
    <includes>
      <include name="IDR_ACCESSIBILITY_HTML"
        file="browser/resources/accessibility/accessibility.html"
        flattenhtml="true"
        allowexternalscript="true"
        type="BINDATA" />
    </includes>
  </release>
<grit>
```

可见，一个 grd 文件的根节点是 grit 节点，grit 节点包括 outputs、translations、release 三个子节点。

其中 outputs 表示目标生成文件，可以为每种语言指定生成的目标文件。相应的，translations 表示每种语言的翻译文件。而 release 节点则是字符串资源 message 以及文件资源 include 等，其中 message 资源的 name 是 IDS_开头，而 include 资源的 name 是 IDR_开头。

每种语言都对应一个 xtb 翻译文件，里面是为每个 id 的字符串资源指定了对应语言的翻译文本。

1.1 outputs

outputs 为每种语言指定了一个输出文件，典型示例如下：

```
  <outputs>
    <output filename="grit/locale_settings.h" type="rc_header">
      <emit emit_type='prepend'></emit>
    </output>
    <output filename="locale_settings_en-US.pak"
           type="data_package" lang="en" />
    <output filename="platform_locale_settings_zh-CN.pak"
           type="data_package" lang="zh-CN" />
  </outputs>
```

其中，每个 output 指定了一个输出文件，每个 output 包括 filename、type 属性，type 有 rc_header、rc_all、data_package；而 data_package 还包含 lang 属性。

1.2 translations

translations 为每种语言指定了一个翻译文件，典型示例如下：

```
  <translations>
    <file path="platform_locale_settings/locale_settings_win_zh-CN.xtb"
        lang="zh-CN" />
  </translations>
```

file 节点有两个属性，path 和 lang。path 指定翻译文件，翻译文件是一个 xtb 后缀的 xml 文件，lang 是语言名称。

1.3 release

release 节点包含字符串资源和文件资源，字符串资源包含在 messages 节点下，而文件资源则包含在 includes 节点下，示例如下：

```
<release>
    <messages fallback_to_english="true">
      <message name="IDS_FIXED_FONT_FAMILY" use_name_for_id="true">
            Courier New
      </message>
    </messages>
    <includes>
      <include name="IDR_ACCESSIBILITY_HTML"
        file="browser/resources/accessibility/accessibility.html"
        flattenhtml="true"
        allowexternalscript="true"
        type="BINDATA" />
    </includes>
  </release>
```

1.3.1 message

每个 message 节点代表一个字符串资源，有 name、use_name_for_id 两个属性。name 以 IDS 开头。另外，message 的值可以使用站位符：

```
<message name="IDS_FORM_FILE_MULTIPLE_UPLOAD"
  desc="text to display next to file buttons in HTML forms when 2 or more files are selected for uploading. This is not used for a case that just 1 file is selected.">
        <ph name="NUMBER_OF_FILES">$1<ex>3</ex></ph> files
</message>
```

此处使用<ph name="NUMBER_OF_FILES">$1<ex>3</ex></ph> 做占位符，在使用 l10n_util::GetStringUTF16 获取文本资源时，可以传入最多 4 个参数替换$1 到$4 的占位符。

1.3.2 include

每个 include 节点代表一个文件资源，有 name、file、type 等节点。如果是 html、js、css 文件，还可以拥有 flattenhtml 属性，如果是 html、js 文件，有 allowexternalscript 属性。

2 XTB 文件

xtb 文件可以看做是 grd 的翻译文件，大概格式如下：

```
<? xml version="1.0"    ? >
<!DOCTYPE translationbundle>
<translationbundle lang="zh-CN">
<translation id="6676384891291319759">访问互联网</translation>
<translation id="6373523479360886564">确定要卸载 Chromium 吗？</translation>
<translation id="5065199687811594072">您希望 Chromium 保存该信用卡信息以便填写网络表单吗？</translation>
<translation id="6510925080656968729">卸载 Chromium</translation>
</ translationbundle>
```

其中每个 id 唯一代表了 grd 文件里的某个 id。TODO：搞清楚 id 是如何生成的，似乎 xtb 文件是先通过脚本系统自动生成，然后再编辑。

3 GRIT 脚本系统

grit 是 Google Resource and Internationalization Tool 的缩写，通过 Google 资源和国际化工具，将 grd 和 xtb 文件生成字符串 id 头文件和包含字符串的 rc 文件，将 grd 的文件资源打包成 pak 文件。

src/chrome/tools/check_grd_for_unused_strings.py
用来检查未使用的字符串。

4 PAK 文件

根据定义的 grd、xtb 文件，以及资源文件，通过 grid 脚本系统生成对应的 c++字符串资源头文件，rc 文件以及 pak 文件。下面列举几个不同层的生成 pak 文件的工程：
net_resources.vcxproj
ui_resources.vcxproj
webkit_resources.vcxproj
content_resources.vcxproj
content_shell_resources.vcxproj

4.1 生成 PAK 文件

以 net_resource.vcxproj 为例，该项目包含一个文件 resource_ids 文件，该文件位于：
"chromium\src\tools\gritsettings\resource_ids"
这是 grit 系统的 id 分配文件，整个 chromium 的所有资源 id 分配都在这里定义。

```
#
# This file is used to assign starting resource ids for resources and strings
# used by Chromium.  This is done to ensure that resource ids are unique
# across all the grd files.  If you are adding a new grd file, please add
# a new entry to this file.
#
# The first entry in the file, SRCDIR, is special: It is a relative path from
# this file to the base of your checkout.
#
# http://msdn.microsoft.com/en-us/library/t2zechd4(VS.71).aspx says that the
# range for IDR_ is 1 to 28,671 and the range for IDS_ is 1 to 32,767 and
# common convention starts practical use of IDs at 100 or 101.
{
  "SRCDIR": "../..",

  "chrome/browser/browser_resources.grd": {
    "includes": [500],
    "structures": [750],
  },
  "chrome/browser/resources/component_extension_resources.grd": {
    "includes": [1000],
    "structures": [1450],
  },
  "chrome/browser/resources/net_internals_resources.grd": {
    "includes": [1500],
  },
  "ui/webui/resources/webui_resources.grd": {
    "includes": [2000],
    "structures": [2200],
  },
  ...
```

该文件的属性->Custom Build Tools 页面指定了编译脚本：

4.1.1 Command Line

```
call call python "..\tools\grit\grit.py" "-i" "base\net_resources.grd" "build"
"-f" "..\tools\gritsettings\resource_ids"
"-o" "$(OutDir)obj\global_intermediate\net"
"-D" "_chromium"
"-E" "CHROMIUM_BUILD=chromium"
"-D" "toolkit_views"
"-D" "remoting"
"-D" "enable_extensions"
"-D" "enable_printing"
"-D" "enable_themes"
"-D" "enable_app_list"
"-D" "enable_settings_app"
"-D" "enable_google_now"
"-D" "use_concatenated_impulse_responses"
"-D" "enable_webrtc" "-D" "enable_mdns"
```

4.1.2 Output

```
$(OutDir)obj\global_intermediate\net\grit\net_resources.h
$(OutDir)obj\global_intermediate\net\net_resources.pak
$(OutDir)obj\global_intermediate\net\net_resources.rc
```

4.1.3 Additional Dependencies

```
..\tools\grit\grit\format\policy_templates\PRESUBMIT.py
..\tools\grit\grit\format\html_inline_unittest.py
..\tools\grit\grit\tool\resize.py
..\tools\grit\grit\gather\chrome_html_unittest.py
..\tools\grit\grit\lazy_re_unittest.py
..\tools\grit\grit\__init__.py
..\tools\grit\grit\tclib_unittest.py
..\tools\grit\grit\exception.py
base\net_resources.grd
..\tools\grit\grit\gather\txt.py
..\tools\grit\grit\format\js_map_format_unittest.py
..\tools\grit\grit\pseudo_rtl.py
..\tools\grit\grit\shortcuts.py
..\tools\grit\grit\format\policy_templates\writers\plist_strings_writer.py
..\tools\grit\grit\clique_unittest.py
base\dir_header.html
..\tools\grit\grit\format\policy_templates\writers\xml_writer_base_unittest.py
..\tools\grit\grit\node\variant.py
..\tools\grit\grit\format\resource_map_unittest.py
..\tools\grit\grit\format\chrome_messages_json_unittest.py
..\tools\grit\grit\gather\rc_unittest.py
..\tools\grit\grit\tool\test.py
..\tools\grit\grit\node\misc_unittest.py
..\tools\grit\grit\format\policy_templates\writers\json_writer.py
..\tools\grit\grit_info.py
..\tools\grit\grit\clique.py
..\tools\grit\grit\tool\preprocess_interface.py
..\tools\grit\grit\pseudo.py
..\tools\grit\grit\gather\igoogle_strings_unittest.py
..\tools\grit\grit\format\rc_header_unittest.py
..\tools\grit\grit\format\policy_templates\writers\admx_writer_unittest.py
..\tools\grit\grit\format\js_map_format.py
..\tools\grit\grit\format\policy_templates\writers\adm_writer.py
..\tools\grit\grit\format\policy_templates\writers\xml_formatted_writer.py
..\tools\grit\grit\format\policy_templates\writers\plist_writer_unittest.py
..\tools\grit\grit\format\policy_templates\writers\admx_writer.py
..\tools\grit\grit\extern\BogoFP.py
..\tools\grit\grit\format\data_pack.py
..\tools\grit\grit\format\policy_templates\writers\adm_writer_unittest.py
..\tools\grit\grit\format\policy_templates\writers\writer_unittest_common.py
..\tools\grit\grit\format\policy_templates\template_formatter.py
```

```
..\tools\grit\grit\gather\json_loader.py
..\tools\grit\grit\tool\menu_from_parts.py
..\tools\grit\grit\gather\muppet_strings.py
..\tools\grit\grit\format\policy_templates\policy_template_generator_unittest.py
..\tools\grit\grit\gather\tr_html_unittest.py
..\tools\grit\grit\node\include.py
..\tools\grit\grit\node\message_unittest.py
..\tools\grit\grit\gather\rc.py
..\tools\grit\grit\tool\rc2grd.py
..\tools\grit\grit\node\structure_unittest.py
..\tools\grit\grit\format\policy_templates\__init__.py
..\tools\grit\grit\tool\buildinfo.py
..\tools\grit\grit\gather\skeleton_gatherer.py
..\tools\grit\grit\shortcuts_unittests.py
..\tools\grit\grit\format\data_pack_unittest.py
..\tools\grit\grit\gather\interface.py
..\tools\grit\grit\tool\toolbar_postprocess.py
..\tools\grit\grit\format\policy_templates\writers\template_writer_unittest.py
..\tools\grit\grit\node\custom\filename_unittest.py
..\tools\grit\grit\format\policy_templates\writers\plist_helper.py
..\tools\grit\grit\node\misc.py
..\tools\grit\grit\format\policy_templates\writers\plist_writer.py
..\tools\grit\grit\tool\transl2tc.py
..\tools\grit\grit\extern\__init__.py
..\tools\grit\grit\node\message.py
..\tools\grit\grit\tool\android2grd.py
..\tools\grit\grit\format\policy_templates\writers\reg_writer.py
..\tools\grit\grit\format\html_inline.py
..\tools\grit\grit\extern\FP.py
..\tools\grit\grit\tool\diff_structures.py
..\tools\grit\grit\gather\admin_template.py
..\tools\grit\grit\grit_runner.py
..\tools\grit\grit\format\c_format.py
..\tools\grit\grit\gather\txt_unittest.py
..\tools\grit\grit\format\policy_templates\writers\adml_writer_unittest.py
..\tools\grit\grit\format\android_xml_unittest.py
..\tools\grit\grit\tool\newgrd.py
..\tools\grit\grit\node\custom\filename.py
..\tools\grit\grit\format\resource_map.py
..\tools\grit\grit\format\policy_templates\writers\__init__.py
..\tools\grit\grit\tool\unit.py
..\tools\grit\grit\util.py
..\tools\grit\grit\format\policy_templates\writers\template_writer.py
..\tools\grit\grit\format\policy_templates\writers\mock_writer.py
```

```
..\tools\grit\grit\tool\count.py
..\tools\grit\grit\tool\android2grd_unittest.py
..\tools\grit\grit\lazy_re.py
..\tools\grit\grit\format\rc.py
..\tools\grit\grit\node\structure.py
..\tools\grit\grit\node\io_unittest.py
..\tools\grit\grit\format\policy_templates\writers\doc_writer_unittest.py
..\tools\grit\grit\grd_reader.py
..\tools\grit\PRESUBMIT.py
..\tools\grit\grit\test_suite_all.py
..\tools\grit\grit\xtb_reader.py
..\tools\grit\grit\format\policy_templates\writers\plist_strings_writer_unittest.py
..\tools\grit\grit\util_unittest.py
..\tools\grit\grit\format\policy_templates\policy_template_generator.py
..\tools\grit\grit\tool\xmb_unittest.py
..\tools\grit\grit\gather\regexp.py
..\tools\grit\grit\tool\toolbar_preprocess.py
..\tools\grit\grit\format\policy_templates\writers\doc_writer.py
..\tools\grit\grit\grd_reader_unittest.py
..\tools\grit\grit\node\base_unittest.py
..\tools\grit\grit\tool\postprocess_interface.py
..\tools\grit\grit\format\repack.py
..\tools\grit\grit\tool\__init__.py
..\tools\grit\grit\gather\tr_html.py
..\tools\grit\grit\extern\tclib.py
..\tools\grit\grit\format\android_xml.py
..\tools\grit\grit\tool\xmb.py
..\tools\grit\grit\format\__init__.py
..\tools\grit\grit\grit_runner_unittest.py
..\tools\grit\grit\format\chrome_messages_json.py
..\tools\grit\grit\node\include_unittest.py
..\tools\grit\grit\tool\transl2tc_unittest.py
..\tools\grit\grit\format\policy_templates\writers\reg_writer_unittest.py
..\tools\grit\grit\format\policy_templates\writers\adml_writer.py
..\tools\grit\grit\gather\igoogle_strings.py
..\tools\grit\grit\xtb_reader_unittest.py
..\tools\grit\grit\gather\policy_json_unittest.py
..\tools\grit\grit\gather\chrome_scaled_image.py
..\tools\grit\grit\format\policy_templates\writers\json_writer_unittest.py
..\tools\grit\grit\tool\postprocess_unittest.py
..\tools\grit\grit\gather\chrome_scaled_image_unittest.py
..\tools\grit\grit\format\c_format_unittest.py
..\tools\grit\grit\scons.py
..\tools\grit\grit\gather\muppet_strings_unittest.py
```

```
..\tools\grit\grit\constants.py
..\tools\grit\grit\gather\admin_template_unittest.py
..\tools\grit\grit\gather\chrome_html.py
..\tools\grit\grit\node\mapping.py
..\tools\grit\grit\gather\__init__.py
..\tools\grit\grit\node\empty.py
..\tools\grit\grit\tclib.py
..\tools\grit\grit\node\__init__.py
..\tools\grit\grit\gather\policy_json.py
..\tools\grit\grit\tool\rc2grd_unittest.py
..\tools\grit\grit\node\custom\__init__.py
..\tools\grit\grit\tool\buildinfo_unittest.py
..\tools\grit\grit\tool\interface.py
..\tools\grit\grit.py
..\tools\grit\grit\tool\preprocess_unittest.py
..\tools\grit\grit\format\rc_unittest.py
..\tools\grit\grit\tool\build_unittest.py
..\tools\grit\grit\node\base.py
..\tools\grit\grit\node\io.py
..\tools\grit\grit\pseudo_unittest.py
..\tools\grit\grit\tool\build.py
..\tools\grit\grit\format\rc_header.py
..\tools\grit\grit\format\policy_templates\writer_configuration.py
```

4.2 合并 PAK 文件

Pak 文件也可以合并，以 content_shell_pak.vcxproj 为例，该项目合成了 content、net、ui、webkit 等下层的 pak 文件。该项目包含 repack.py 文件，该文件位置如下：
"chromium\src\tools\grit\grit\format\repack.py"
该文件的属性->Custom Build Tools 页面指定了编译脚本：

4.2.1 Command Line

```
call call "$(ProjectDir)..\third_party\cygwin\setup_env.bat" &&

set CYGWIN=nontsec &&

set OUTDIR=$(OutDir) &&

bash -c "\"python\" \"../tools/grit/grit/format/repack.py\"

\"`cygpath -m \\\"${OUTDIR}\\\"`/content_shell.pak\"

\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/content/content_resources.pak\"

\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/content/browser/tracing/tracing_resources.pak\"

\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/content/shell_resources.pak\"

\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/net/net_resources.pak\"
```

```
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/ui/app_locale_settings/app_locale_settings_en-US.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/ui/ui_resources/ui_resources_100_percent.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/ui/ui_resources/webui_resources.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/ui/ui_strings/ui_strings_en-US.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/webkit/devtools_resources.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/webkit/blink_resources.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/webkit/webkit_resources_100_percent.pak\"
\"`cygpath -m \\\"${OUTDIR}\\\"`obj/global_intermediate/webkit/webkit_strings_en-US.pak\""
```

4.2.2 Output

```
$(OutDir)\content_shell.pak
```

4.2.3 Additional Dependencies

```
$(OutDir)obj\global_intermediate\ui\app_locale_settings\app_locale_settings_en-US.pak
$(OutDir)obj\global_intermediate\net\net_resources.pak
$(OutDir)obj\global_intermediate\webkit\devtools_resources.pak
$(OutDir)obj\global_intermediate\webkit\webkit_resources_100_percent.pak
$(OutDir)obj\global_intermediate\webkit\blink_resources.pak
$(OutDir)obj\global_intermediate\content\shell_resources.pak
$(OutDir)obj\global_intermediate\ui\ui_resources\ui_resources_100_percent.pak
$(OutDir)obj\global_intermediate\ui\ui_resources\webui_resources.pak
$(OutDir)obj\global_intermediate\webkit\webkit_strings_en-US.pak
$(OutDir)obj\global_intermediate\content\browser\tracing\tracing_resources.pak
$(OutDir)obj\global_intermediate\ui\ui_strings\ui_strings_en-US.pak
$(OutDir)obj\global_intermediate\content\content_resources.pak
```

5 资源加载

5.1 ui::ResourceBundle

ResourceBundle 是 Google 资源的加载器，该类位于：

chromium\src\ui\base\resource\resource_bundle.h
chromium\src\ui\base\resource\resource_bundle.cc

5.1.1 初始化函数

```
// Initialize the ResourceBundle for this process. Does not take ownership of
// the |delegate| value. Returns the language selected.
// NOTE: Mac ignores this and always loads up resources for the language
// defined by the Cocoa UI (i.e., NSBundle does the language work).
//
// TODO(sergeyu): This method also loads common resources (i.e. chrome.pak).
// There is no way to specify which resource files are loaded, i.e. names of
// the files are hardcoded in ResourceBundle. Fix it to allow to specify which
// files are loaded (e.g. add a new method in Delegate).
static std::string InitSharedInstanceWithLocale(
    const std::string& pref_locale, Delegate* delegate);

// Same as InitSharedInstanceWithLocale(), but loads only localized resources,
// without default resource packs.
static std::string InitSharedInstanceLocaleOnly(
    const std::string& pref_locale, Delegate* delegate);

// Initialize the ResourceBundle using given file. The second argument
// controls whether or not ResourceBundle::LoadCommonResources is called.
// This allows the use of this function in a sandbox without local file
// access (as on Android).
static void InitSharedInstanceWithPakFile(
    base::PlatformFile file, bool should_load_common_resources);

// Delete the ResourceBundle for this process if it exists.
static void CleanupSharedInstance();

// Returns true after the global resource loader instance has been created.
static bool HasSharedInstance();

// Return the global resource loader instance.
static ResourceBundle& GetSharedInstance();
```

初始化的逻辑是：
1、调用私有函数 LoadCommonResources 加载平台相关的 chrome_xxx.pak
2、调用私有函数 LoadLocaleResources 加载本地 pak 文件。

5.1.2 添加 pak 资源文件函数

```
// Registers additional data pack files with this ResourceBundle.  When
// looking for a DataResource, we will search these files after searching the
// main module. |path| should be the complete path to the pack file if known
// or just the pack file name otherwise (the delegate may optionally override
// this value). |scale_factor| is the scale of images in this resource pak
// relative to the images in the 1x resource pak. This method is not thread
// safe! You should call it immediately after calling InitSharedInstance.
void AddDataPackFromPath(const base::FilePath& path,
                         ScaleFactor scale_factor);

// Same as above but using an already open file.
void AddDataPackFromFile(base::PlatformFile file, ScaleFactor scale_factor);

// Same as AddDataPackFromPath but does not log an error if the pack fails to
// load.
void AddOptionalDataPackFromPath(const base::FilePath& path,
                                 ScaleFactor scale_factor);
```

5.1.3 加载资源函数

```
// Gets image with the specified resource_id from the current module data.
// Returns a pointer to a shared instance of gfx::ImageSkia. This shared
// instance is owned by the resource bundle and should not be freed.
// TODO(pkotwicz): Make method return const gfx::ImageSkia*
//
// NOTE: GetNativeImageNamed is preferred for cross-platform gfx::Image use.
gfx::ImageSkia* GetImageSkiaNamed(int resource_id);

// Gets an image resource from the current module data. This will load the
// image in Skia format by default. The ResourceBundle owns this.
gfx::Image& GetImageNamed(int resource_id);

// Similar to GetImageNamed, but rather than loading the image in Skia format,
// it will load in the native platform type. This can avoid conversion from
// one image type to another. ResourceBundle owns the result.
//
// Note that if the same resource has already been loaded in GetImageNamed(),
// gfx::Image will perform a conversion, rather than using the native image
// loading code of ResourceBundle.
```

```
//
// If |rtl| is RTL_ENABLED then the image is flipped in RTL locales.
gfx::Image& GetNativeImageNamed(int resource_id, ImageRTL rtl);

// Same as GetNativeImageNamed() except that RTL is not enabled.
gfx::Image& GetNativeImageNamed(int resource_id);

// Loads the raw bytes of a scale independent data resource.
base::RefCountedStaticMemory* LoadDataResourceBytes(int resource_id) const;

// Loads the raw bytes of a data resource nearest the scale factor
// |scale_factor| into |bytes|, without doing any processing or
// interpretation of the resource. Use ResourceHandle::SCALE_FACTOR_NONE
// for scale independent image resources (such as wallpaper).
// Returns NULL if we fail to read the resource.
base::RefCountedStaticMemory* LoadDataResourceBytesForScale(
    int resource_id,
    ScaleFactor scale_factor) const;

// Return the contents of a scale independent resource in a
// StringPiece given the resource id
base::StringPiece GetRawDataResource(int resource_id) const;

// Return the contents of a resource in a StringPiece given the resource id
// nearest the scale factor |scale_factor|.
// Use ResourceHandle::SCALE_FACTOR_NONE for scale independent image resources
// (such as wallpaper).
base::StringPiece GetRawDataResourceForScale(int resource_id,
                                             ScaleFactor scale_factor) const;

// Get a localized string given a message id.  Returns an empty
// string if the message_id is not found.
string16 GetLocalizedString(int message_id);

// Returns the font list for the specified style.
const gfx::FontList& GetFontList(FontStyle style);

// Returns the font for the specified style.
const gfx::Font& GetFont(FontStyle style);
```

5.2 ui::ResourceBundle::Delegate

该类是资源加载的委托类，控制资源的加载。

5.3 content::ContentMainDelegate:: InitializeResourceBundle

content::ContentMainDelegate 接口类的 InitializeResourceBundle 方法里需要初始化 ResourceBundle,初始化该类可以指定 ui::ResourceBundle::Delegate 的实现者。

ResourceBundle::InitSharedInstanceWithLocale，先加载Chrome的pak文件，主要是不同设备和dpi下的图片资源。之后再加载Locale的PAK文件，包括不同语言的文字定义。

ResourceBundle::InitSharedInstanceLocaleOnly，只加载Locale文件，不加载Chrome的pak文件。

5.4 l10n_util::GetStringUTF8/ l10n_util::GetStringUTF16

下面两个函数直接获取指定 id 的字符串

```
std::string GetStringUTF8(int message_id) {
  return UTF16ToUTF8(GetStringUTF16(message_id));
}

string16 GetStringUTF16(int message_id) {
  ResourceBundle& rb = ResourceBundle::GetSharedInstance();
  string16 str = rb.GetLocalizedString(message_id);
  AdjustParagraphDirectionality(&str);

  return str;
}
```

可见这两个函数最终转调用ResourceBundle::GetLocalizedString方法。

下面这个函数获取指定id字符串，并使用额外的参数替换目标字符串里的$i占位符。

```
string16 GetStringFUTF16(int message_id,
                         const std::vector<string16>& replacements,
                         std::vector<size_t>* offsets) {
  // TODO(tc): We could save a string copy if we got the raw string as
  // a StringPiece and were able to call ReplaceStringPlaceholders with
  // a StringPiece format string and string16 substitution strings.  In
  // practice, the strings should be relatively short.
  ResourceBundle& rb = ResourceBundle::GetSharedInstance();
  const string16& format_string = rb.GetLocalizedString(message_id);
  string16 formatted = ReplaceStringPlaceholders(format_string, replacements,
                                                 offsets);
  AdjustParagraphDirectionality(&formatted);

  return formatted;
}
```

下面是一组重载函数

```
std::string GetStringFUTF8(int message_id,
                           const string16& a) {
  return UTF16ToUTF8(GetStringFUTF16(message_id, a));
}

std::string GetStringFUTF8(int message_id,
                           const string16& a,
                           const string16& b) {
  return UTF16ToUTF8(GetStringFUTF16(message_id, a, b));
}

std::string GetStringFUTF8(int message_id,
                           const string16& a,
                           const string16& b,
                           const string16& c) {
  return UTF16ToUTF8(GetStringFUTF16(message_id, a, b, c));
}

std::string GetStringFUTF8(int message_id,
                           const string16& a,
                           const string16& b,
                           const string16& c,
                           const string16& d) {
  return UTF16ToUTF8(GetStringFUTF16(message_id, a, b, c, d));
}

string16 GetStringFUTF16(int message_id,
                         const string16& a) {
  std::vector<string16> replacements;
  replacements.push_back(a);
  return GetStringFUTF16(message_id, replacements, NULL);
}

string16 GetStringFUTF16(int message_id,
                         const string16& a,
                         const string16& b) {
  return GetStringFUTF16(message_id, a, b, NULL);
}

string16 GetStringFUTF16(int message_id,
                         const string16& a,
                         const string16& b,
                         const string16& c) {
  std::vector<string16> replacements;
  replacements.push_back(a);
  replacements.push_back(b);
  replacements.push_back(c);
  return GetStringFUTF16(message_id, replacements, NULL);
}

string16 GetStringFUTF16(int message_id,
                         const string16& a,
                         const string16& b,
                         const string16& c,
                         const string16& d) {
  std::vector<string16> replacements;
  replacements.push_back(a);
  replacements.push_back(b);
  replacements.push_back(c);
  replacements.push_back(d);
  return GetStringFUTF16(message_id, replacements, NULL);
}

string16 GetStringFUTF16(int message_id,
                         const string16& a,
                         const string16& b,
                         const string16& c,
                         const string16& d,
                         const string16& e) {
  std::vector<string16> replacements;
  replacements.push_back(a);
  replacements.push_back(b);
  replacements.push_back(c);
  replacements.push_back(d);
  replacements.push_back(e);
  return GetStringFUTF16(message_id, replacements, NULL);
}

string16 GetStringFUTF16(int message_id, const string16& a, size_t* offset) {
  DCHECK(offset);
  std::vector<size_t> offsets;
  std::vector<string16> replacements;
  replacements.push_back(a);
  string16 result = GetStringFUTF16(message_id, replacements, &offsets);
  DCHECK(offsets.size() == 1);
  *offset = offsets[0];
  return result;
}

string16 GetStringFUTF16(int message_id,
                         const string16& a,
                         const string16& b,
                         std::vector<size_t>* offsets) {
  std::vector<string16> replacements;
  replacements.push_back(a);
  replacements.push_back(b);
  return GetStringFUTF16(message_id, replacements, offsets);
}

string16 GetStringFUTF16Int(int message_id, int a) {
  return GetStringFUTF16(message_id, UTF8ToUTF16(base::IntToString(a)));
}

string16 GetStringFUTF16Int(int message_id, int64 a) {
  return GetStringFUTF16(message_id, UTF8ToUTF16(base::Int64ToString(a)));
}
```

其中ReplaceStringPlaceHodlers方法最终调用如下函数进行替换：

```
template<class FormatStringType, class OutStringType>
OutStringType DoReplaceStringPlaceholders(const FormatStringType& format_string,
    const std::vector<OutStringType>& subst, std::vector<size_t>* offsets) {
  size_t substitutions = subst.size();

  size_t sub_length = 0;
  for (typename std::vector<OutStringType>::const_iterator iter = subst.begin();
       iter != subst.end(); ++iter) {
    sub_length += iter->length();
  }

  OutStringType formatted;
  formatted.reserve(format_string.length() + sub_length);

  std::vector<ReplacementOffset> r_offsets;
```

```
for (typename FormatStringType::const_iterator i = format_string.begin();
     i != format_string.end(); ++i) {
  if ('$' == *i) {
    if (i + 1 != format_string.end()) {
      ++i;
      DCHECK('$' == *i || '1' <= *i) << "Invalid placeholder: " << *i;
      if ('$' == *i) {
        while (i != format_string.end() && '$' == *i) {
          formatted.push_back('$');
          ++i;
        }
        --i;
      } else {
        uintptr_t index = 0;
        while (i != format_string.end() && '0' <= *i && *i <= '9') {
          index *= 10;
          index += *i - '0';
          ++i;
        }
        --i;
        index -= 1;
        if (offsets) {
          ReplacementOffset r_offset(index,
              static_cast<int>(formatted.size()));
          r_offsets.insert(std::lower_bound(r_offsets.begin(),
                                            r_offsets.end(),
                                            r_offset,
                                            &CompareParameter),
                           r_offset);
        }
        if (index < substitutions)
          formatted.append(subst.at(index));
      }
    }
  } else {
    formatted.push_back(*i);
  }
}
if (offsets) {
  for (std::vector<ReplacementOffset>::const_iterator i = r_offsets.begin();
       i != r_offsets.end(); ++i) {
    offsets->push_back(i->offset);
  }
}
```

```
  return formatted;
}
```

而ResourceBundle::GetLocalizedString函数如下：

```
string16 ResourceBundle::GetLocalizedString(int message_id) {
  string16 string;
  if (delegate_ && delegate_->GetLocalizedString(message_id, &string))
    return string;

  // Ensure that ReloadLocaleResources() doesn't drop the resources while
  // we're using them.
  base::AutoLock lock_scope(*locale_resources_data_lock_);

  // If for some reason we were unable to load the resources , return an empty
  // string (better than crashing).
  if (!locale_resources_data_.get()) {
    LOG(WARNING) << "locale resources are not loaded";
    return string16();
  }

  base::StringPiece data;
  if (!locale_resources_data_->GetStringPiece(message_id, &data)) {
    // Fall back on the main data pack (shouldn't be any strings here except in
    // unittests).
    data = GetRawDataResource(message_id);
    if (data.empty()) {
      NOTREACHED() << "unable to find resource: " << message_id;
      return string16();
    }
  }

  // Strings should not be loaded from a data pack that contains binary data.
  ResourceHandle::TextEncodingType encoding =
      locale_resources_data_->GetTextEncodingType();
  DCHECK(encoding == ResourceHandle::UTF16 || encoding == ResourceHandle::UTF8)
      << "requested localized string from binary pack file";

  // Data pack encodes strings as either UTF8 or UTF16.
  string16 msg;
  if (encoding == ResourceHandle::UTF16) {
    msg = string16(reinterpret_cast<const char16*>(data.data()),
                   data.length() / 2);
  } else if (encoding == ResourceHandle::UTF8) {
    msg = UTF8ToUTF16(data);
  }
  return msg;
}
```

三行黄色背景代码是获取字符串资源的顺序：

1、 从ResourceBundle::Delegate::GetLocalizedStromg获取字符串，如果存在则返回。

2、 从locale_resources_data_获取字符串，如果不存在则从3获取。

3、 调用GetRawDataResource获取，这个函数内部最终调用data_packs_获取

通过分析，可知，local_resources_data_和data_packs_的类型都是ResourceHandle。在windows平台上不同的是，local_resources_data_初始化时的是DataPack类，而data_packs_反而初始化的是ResourceDataDll类，这两个类都是ResourceHandle的子类。

6 Chromium_resources

chromium_resources.sln是chromium的资源解决方案，最终所需要的pak分别由下面两个项目合成：

- chrome/packed_extra_resoures.vcxproj

1） Debug目录下的chrome.pak、chrome_100_percent.pak chrome_touch_100_percent.pak等

2）Debug\local目录下的各种语言包pak，比如en-US.pak

- chrome/packed_resources.vcxproj
  Debug目录下的resources.pak

chromium_resources下项目输出文件都在临时目录下：$(OutDir)obj\global_intermediate