
Mostly Other People Do the Killing: Changing our stable API incompatibly, revisited

Stephan Bergmann
Red Hat, Inc.

A Socio-Technological Issue

- Incompatible changes to the URE interface
 - UNOIDL entities, C++/Java runtime environment
 - Basis for extensions, macros, core code
- Change ultimately affects people
 - Coordinate with people
- Mitigate disruption through technology

The Social Component



Where we are

- “Never change the stable URE interface”
 - Lead to lots of cruft and frustration
- Since LibreOffice 4: Deliberately break the interface to improve a specific area
 - Only in major/minor versions (4.0, 4.1, 4.2, ...)
 - But broke that for date/time API in 4.1.1
 - Consensus-driven (ESC)
 - Others do this too, now (AOO 4)

What we left behind

- `IllegalArgumentException` is a `RuntimeException`
- `XDocumentInfo` removed
- `DateTime`:
 - unsigned short `HundredthSeconds` to unsigned long `NanoSeconds`
 - unsigned short to short `Year`
 - additional `isUTC` flag
- ...

Who cares?

- `IllegalArgumentException` is a `RuntimeException`:
 - 3 commits mention it
- `XDocumentInfo` removed:
 - 0 commits mention it
- `DateTime` changes:
 - 6 commits mention it
- Plus problems in macros, extensions
 - Mostly only reported after the RC phase

Documentation

- “[API CHANGE]” commit messages
 - `git log --grep='API CHANGE' --oneline | wc -l: 74`
- [http://www.libreoffice.org/download/4-0-new-features-and-fixes/“API Changes”](http://www.libreoffice.org/download/4-0-new-features-and-fixes/API_Changes)
- [http://www.libreoffice.org/download/4-1-new-features-and-fixes/“API Changes”](http://www.libreoffice.org/download/4-1-new-features-and-fixes/API_Changes)
- https://wiki.documentfoundation.org/ReleaseNotes/4.2#API_Changes

Documentation

- <http://api.libreoffice.org/>
- Documentation for the SDK's UNOIDL, C++, Java, ... APIs
- TODO:
 - Make corresponding LO version clear
 - Provide documentation for old LO versions



The Technical Component



Nomenclature

- *Registry format:*
 - New UNOIDL binary .rdb format
 - (compact, no dependency on module store)
 - Legacy binary .rdb format (store-based)
 - Single source file .idl format
 - Source file tree .idl format
 - .../com/sun/star/uno/XInterface.idl
- UNOIDL *entities*, not just *types*

@deprecated

- .idl: `/** @deprecated */`
- available in registries as “deprecated” annotation
- cppumaker .hpp: `SAL_DEPRECATED_INTERNAL`
 - TODO: `SAL_DEPRECATED_INTERNAL`
- TODO: javamaker .class: `@java.lang.Deprecated`
- TODO: Basic, CLI, Python?
- (Also used in native C++ and Java APIs)

@since

- .idl:
 - `/** @since OOo 3.0 */`
 - `/** @since LibreOffice 4.1 */`
- processed by doxygen
- not currently available in registries as annotation
- mostly useful in documentation anyway?
- (Also used in native C++ and Java APIs, processed by doxygen and javadoc)

type_reference

- Check current udkapi, offapi against udkapi/type_reference/udkapi.idl, offapi/type_reference/offapi.idl at build time
 - Catch inadvertent changes to published entities
- Used to be binary .rdb files
 - Hard to maintain
- A single, textual .idl file with all published entities
 - Plus non-published optional base interfaces

Extensions dependencies

- `<LibreOffice-minimal-version value="..."/>`
- `<OpenOffice.org-maximal-version value="..."/>`
- Defining more fine-grained extensions never caught up
- Only helps for extensions, not for macros



A Socio-Technological Issue

- The social component:
 - Incompatible changes affect people
 - Producers of macros and extensions
 - Consumers of macros and extensions
 - Core developers
 - Make affected people aware, early on
 - Affect people in consumable chunks
 - Affect people at reasonable intervals
 - (though XP suggests otherwise...)

A Socio-Technological Issue

- The technical component:
 - Incompatible changes break existing code
 - Macros
 - Extensions
 - Core code
 - Detect incompatibilities before they cause harm
 - (and crashing can be one of the better options)
 - Deliberately rejected writing wrappers
 - We could do better, though

**We come from chaos, you cannot change us
Cannot explain us and that's what makes us**
—John Lydon

