

CCA

Common Component Architecture

What's new in **BABEL** 0.11.0

Gary Kumfert

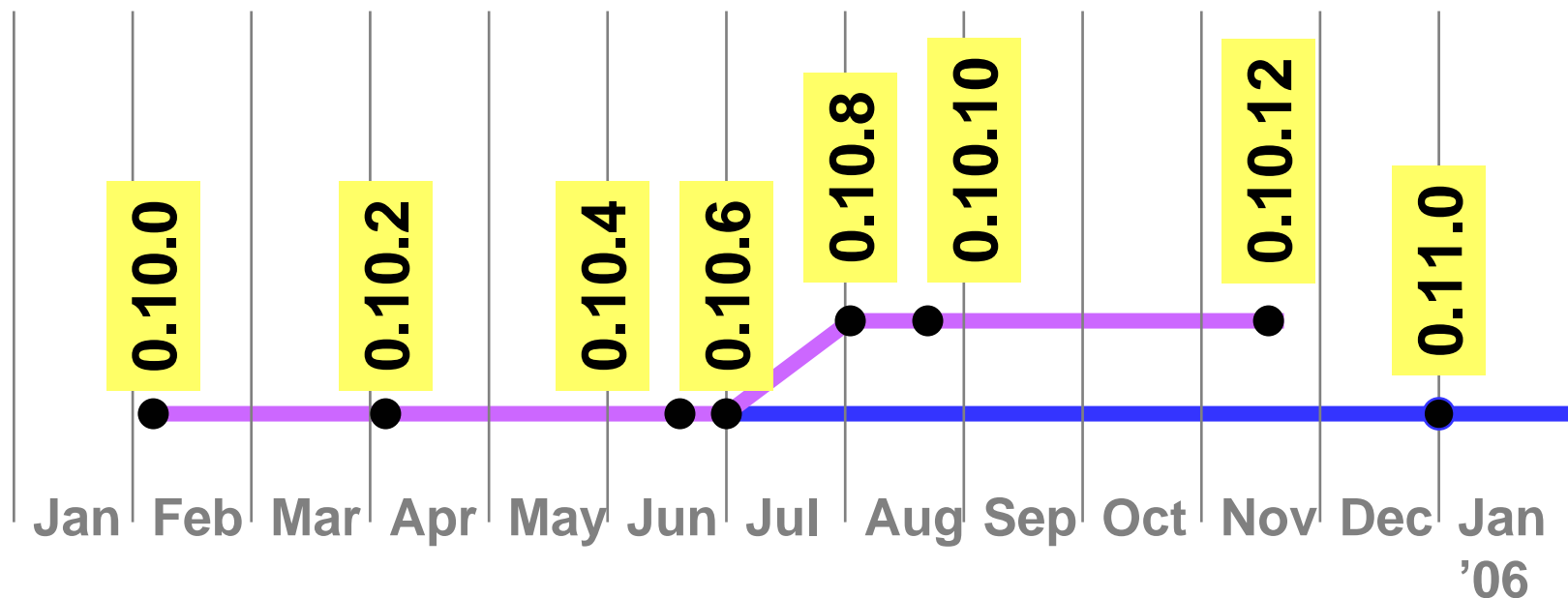
**with Tammy Dahlgren, Thomas Epperly,
James Leek**

This work was performed under the auspices of the U.S. Department of Energy by the University of California, Lawrence Livermore National Laboratory under Contract No. W-7405-Eng-48.

UCRL-PRES-218376

Babel 0.11 represents 6 months of development

- The last 3 Babel releases were off the 0.10.6 branch... not the HEAD



0.11 release is a MAJOR step to 1.0 Release Criteria

- **25 issues identified in December 2003**
- **Some have been added, others deferred over the years**
- **23/27 of criteria are satisfied with the 0.11.0 release**
- **(remaining 4 are “in progress” dealing with parsing & type resolution)**

Two Biggest changes in 0.11 have own technical presentations

- **Introducing RMI -- James Leek**
 - ▶ **Babel is no longer just a language interoperability tool.**

- **Cxx → UCxx Transition – Tom Epperly**
 - ▶ **Officially deprecating a language binding. (You have been warned!)**

RMI had a very long gestation

- **Dec 2000:** First appearance of RMI in a talk to CCA @Gatlinburg. (It is assumed in a broader context of general SIDL interfaces for MxN.)
- **July 2001:** 1st public release of Babel. 0.5.0
- **Dec 2003:** RMI appears as 1.0 Release Criteria
- **July-Dec 2004:** Nija Shi (intern, UC Davis) worked on C Stubs & prototype server in Python
- **Sept-Nov 2004:** Substantial retooling of IOR to support RMI. (r4007—r4122 in svn)

RMI had a very long gestation

- **Feb '05-present: Jim Leek takes on RMI full-time.**
 - ▶ finishes BabelRMI API spec
 - ▶ does TCP/IP-based implementation (distributed in sidlx)
 - ▶ implements access in all language bindings
 - ▶ discovers/resolves all the corner cases (dragging Tom and Gary into serious head scratching mode on several occasions.)

Now that vision becomes reality... almost

- **Vision:** Users would download a new release, and all their Babelized code would be instantly distributed code as well.
- **Reality:** Very close to true, but two big user-visible changes
 - ▶ **runtime exceptions** – a remote method may suffer a network failure
 - ▶ **casting** – initial handle to remote object may be insufficient after downcast

These changes affect ALL users, even with plain in-process Babel.

Added implicit `sidl.RuntimeException` to every SIDL method (not array accessors).

Changed `_cast()` to addRef its return value

- ▶ **1. runtime exceptions: a remote method may suffer a network failure**
- ▶ **2. casting: initial handle to remote object may be insufficient after downcast**

These changes are A Good Thing™ anyway

- **Implicit Exception**

- ▶ Before, many language bindings caught exceptions/errors internally but had no way to signal user code
- ▶ Considering exceptions such as `MethodNotImplemented`, `LanguageSpecific`, etc.

- **`_cast()` bumps ref count:**

- ▶ Makes `_cast()` more consistent with everything else. Every other out parameter and return value already increases ref count.

Changes in C++ support

- **Deprecating original C++ binding**
 - ▶ **Namechange: cxx → dcxx**
 - ▶ **Feature lag: many 0.11 features not available in dcxx**
- **UCxx binding will be new C++ in 1.0**
- **0.11.0 is transition time**
- **UCxx binding had suffix change**
{hh,cc} → {hxx,cxx}

0.11 also includes ALPHAs of important things to come

- New configure flag: `--with-mpi`
- New compilation help: `babel-cc` (and friends)
- New babel option: `-P` (experimental parser)

`./configure --with-mpi`

- **Now:** Only supports MPI-1 + ANSI C
- **Future:**
 - ▶ Will support MPI-1 & MPI-2
 - ▶ Will probe mpicc, mpif90, etc. for all kinds of CFLAGS, LDFLAGS, etc.
- **Why:**
 - ▶ `./configure CC=mpicc` has “issues”
 - ▶ Ties into babel-cc scripts

babel-cc, babel-f90, et. al.

- Follows the mpicc idea... but

- ▶ Also front-end for babel-libtool

- ▶ Also understands multi-languages

```
babel-cc -o prog --with-mpi \  
        --with-cxx *.lo
```

- ▶ Even if you don't use it regularly, the
 --dry-run
 option can be helpful

Experimental Parser

- **Moving from single-pass parser & immediate type resolution**
- **... to multi-pass with lazy resolution**
- **Adding a bunch of syntactic corner cases along the way**
- **HOPING (not promising) to have new one in place for Babel 1.0!**

Acid test for new Parser

- **Diff the entire Babel source tree**
 - ▶ **Only timestamps & metadata should change**
 - ▶ **Same symbol tables & code generators, after all**

Final Note for Babel Modders

- If you make a patch for Babel (includes sumo distros), please...
 - ▶ Create (in the patch) a unique *.txt file in babel/runtime/patches
 - Filename: <14 chars is also the patch name
 - Line 1: short (<40 char) description of patch
 - Line 2: email of primary author or list
 - ▶ Email the patch to babel-dev@llnl.gov
- `babel-config --version-full` will properly report the patch to users

Two Biggest changes in 0.11 have own technical presentations

- **Introducing RMI**
 - James Leek

- **Cxx → UCxx Transition**
 - Thomas Epperly