CHICAGO

INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE

goto; conference

@GOTOCHGO

Conference: April 23-24, 2013 Training: April 25-26, 2013

CHICAGO

INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE

goto; conference

Conference: April 23-24, 2013 Training: April 25-26, 2013

Welcome to GOTO Night Chicago # 1

Embracing Concurrency at Scale – Justin Sheehy, CTO Basho Technologies

Innovation in a Legacy - Dave Thomas, Bedarra Research Labs

Mature Legacy Seeking Sexy New Technology for Fun and Profit

Innovating in a Legacy

Dave Thomas

Bedarra Research Labs YOW/GOTO Conferences,Queensland University of Technology Australia, Carleton University Canada

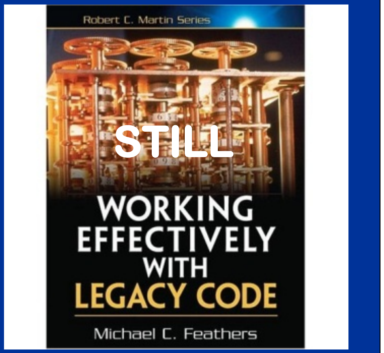
dave@bedarra.com
www.davethomas.net

Once Upon A Legacy

1. Dirty Jobs or Fame & Fortune?
2. No Fear - 1,2,3 Charge!
3. Agile in a Legacy Environment?
4. Time Something Completely Different?
5. Thinking Out of The Legacy Box – Case Studies of Legacy Innovation

“Legacy Code is code that has No tests”

- has insufficient tests
- is older than 3 years
- technical debt written by others . . .
- Is written in a language, tool, platform we don’t know or like
- is difficult to understand or change . . .



New Improved 16th Edition!
(c) 2020

70 – 90 % of Software Budgets are spent on Legacy Systems!

Legacy – *The Big Ball of Mud**

HOW DO WE KNOW YOU CAME TO ELBONIA JUST TO TEACH US CAPITALISM ?
YEAH... MAYBE YOU CAME TO STEAL OUR SECRET PROCESS FOR MAKING MUD !!
Use OO!
HE KNOWS...
WE'LL HAVE TO KILL HIM.
© 1990 United Feature Syndicate, Inc.
DILBERT ® United Feature Syndicate, Inc.
Redistribution in whole or in part prohibited.

* “The Big Ball of Mud” Design Pattern – Brian Foote

Legacy = Big Ball of MUD which has significant business value!!!

Oh My! It is our mess (debt)!?

Our framework adds so much value

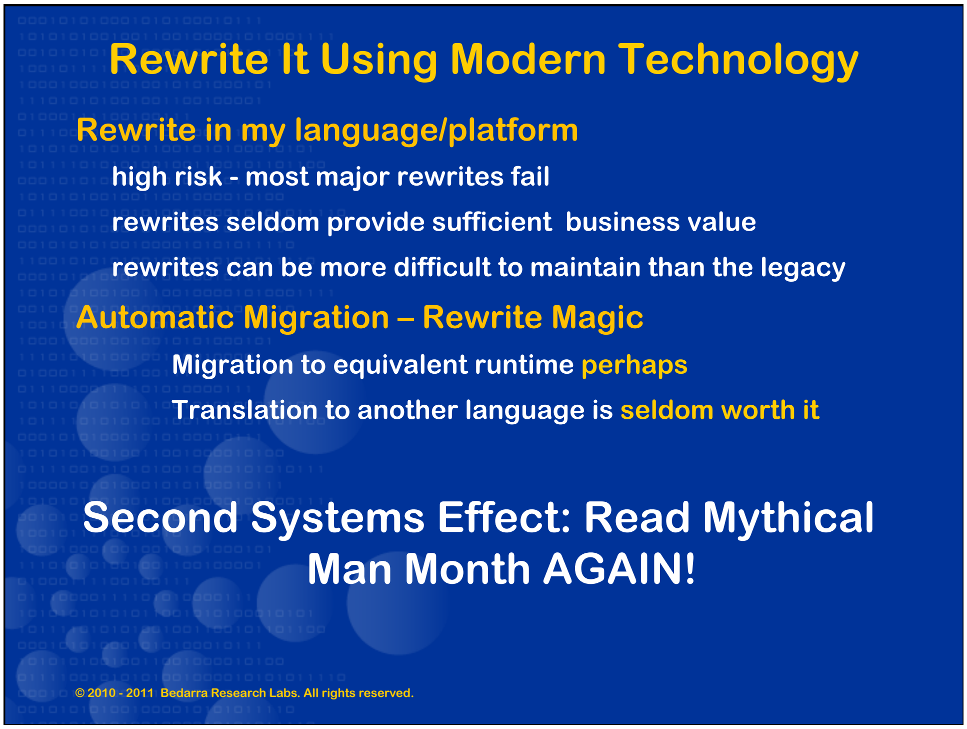
Just add a little glue code and some SOA wrappers

Look we will just have to refactor it later!

Legacy ? No Fear … 1, 2, 3 Charge!

Our vendor! Our consultants! Our outsourcer! Our team has the solution !!!

It will take years and lots of money but we can climb legacy mountain



Rewrite It Using Modern Technology

Rewrite in my language/platform

high risk - most major rewrites fail

rewrites seldom provide sufficient business value

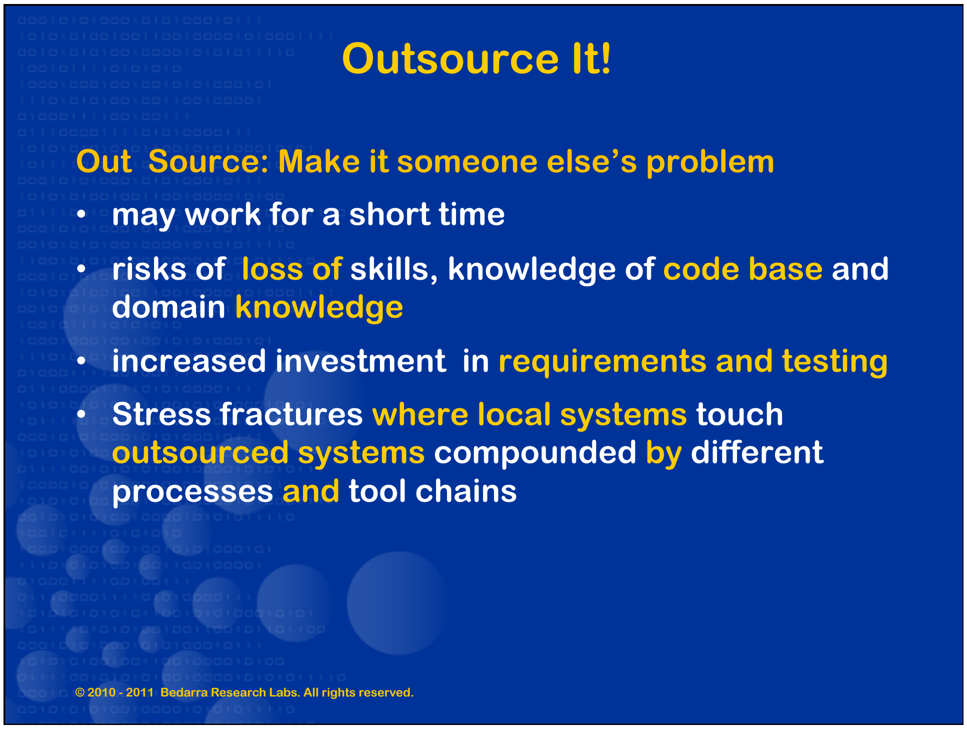
rewrites can be more difficult to maintain than the legacy

Automatic Migration – Rewrite Magic

Migration to equivalent runtime perhaps

Translation to another language is seldom worth it

Second Systems Effect: Read Mythical Man Month AGAIN!



Outsource It!

Out Source: Make it someone else’s problem

- may work for a short time
- risks of loss of skills, knowledge of code base and domain knowledge
- increased investment in requirements and testing
- Stress fractures where local systems touch outsourced systems compounded by different processes and tool chains

Just SOA It!

Just Use SOA: Wrap It Into Services

- Assumes you have interfaces
- Assumes you have tests
- Assumes the service is easily surfaced and encapsulated when in practice the it may be buried in several modules
- Assumes that ESB/XML/Process Server Performance, RMA is acceptable when it often is unknown
- Analysis paralysis between enterprise architects, vendor architects and development teams
  - Which ESB, Process Server …
  - SOAP, REST …
  - XML … JSON …
  - BPM/BPEL …

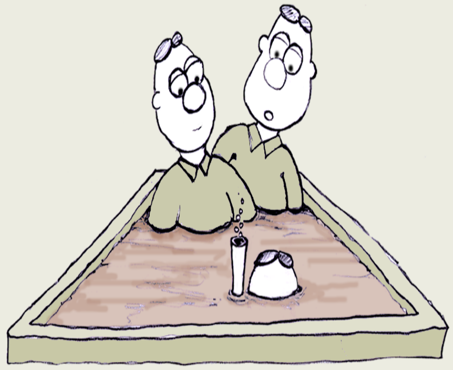
Agile It !

Incrementally Refactor It

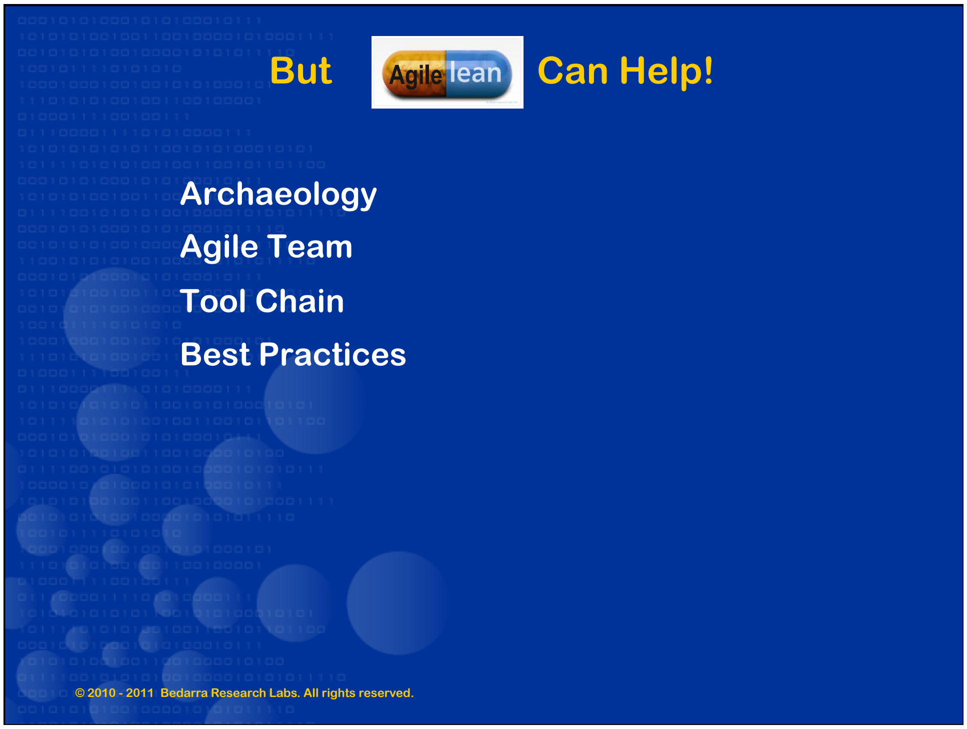
Do you believe in Magic?

- Wizard's work
- Absence of tests significantly increases risk
- Tools are not up to refactoring legacy code bases or databases
- Off the shelf tools may not be available for your legacy language
- Takes too long for the business to sustain it

Refactor THIS Baby!!

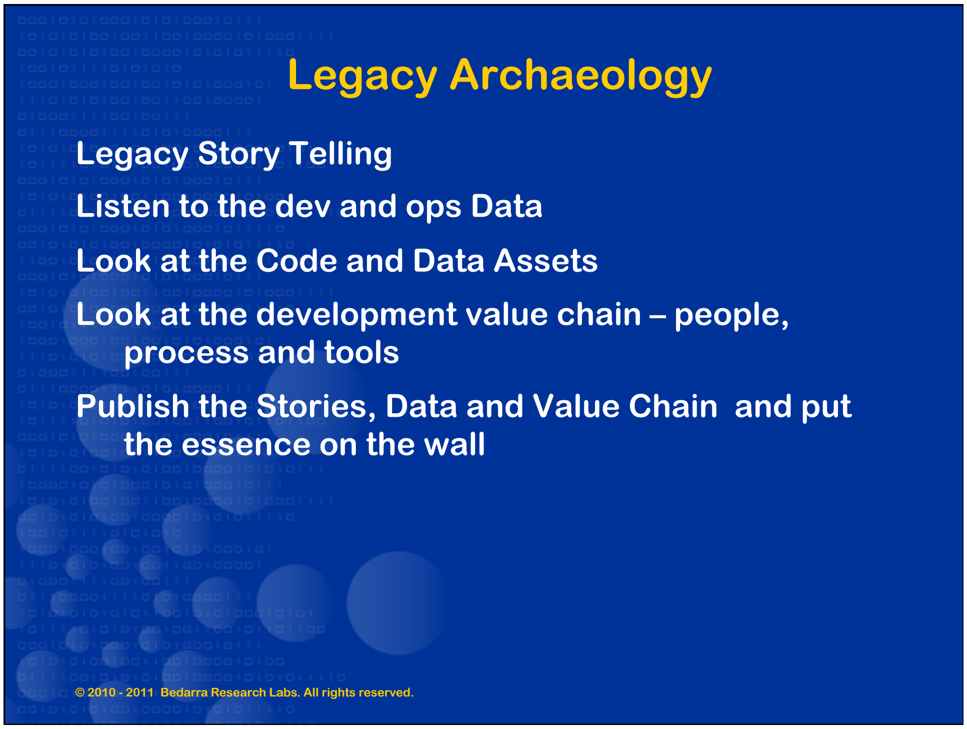


"I don't know about you... but I find these 'legacy' refactoring tools leave a lot to be desired . . .
Yeah, Let's try some dynamite"



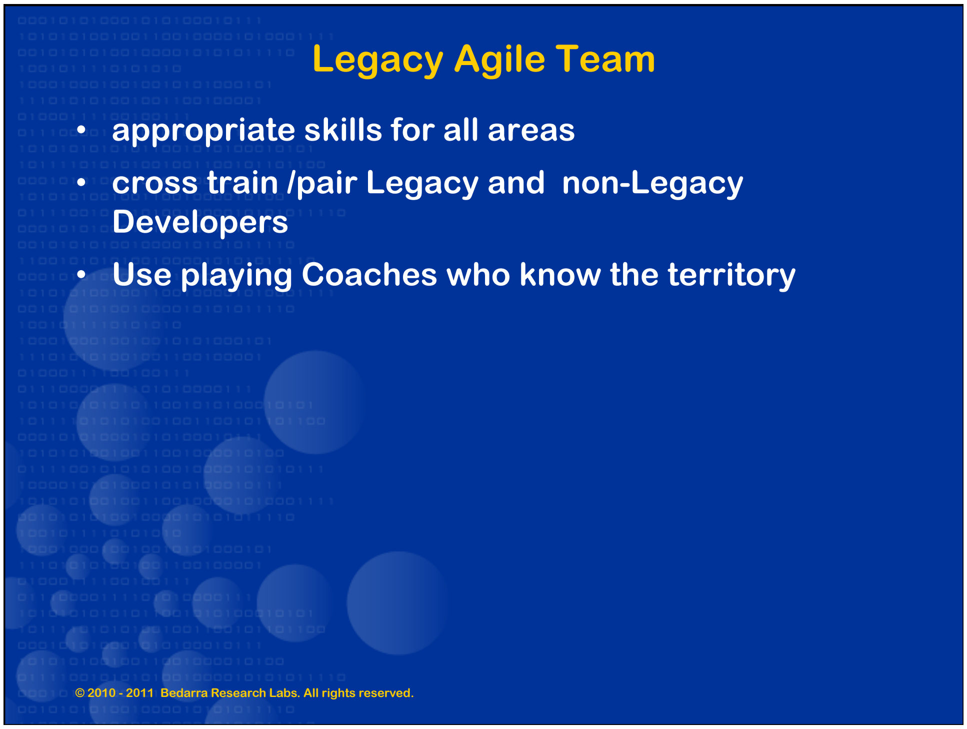
But Agile lean Can Help!

- Archaeology
- Agile Team
- Tool Chain
- Best Practices



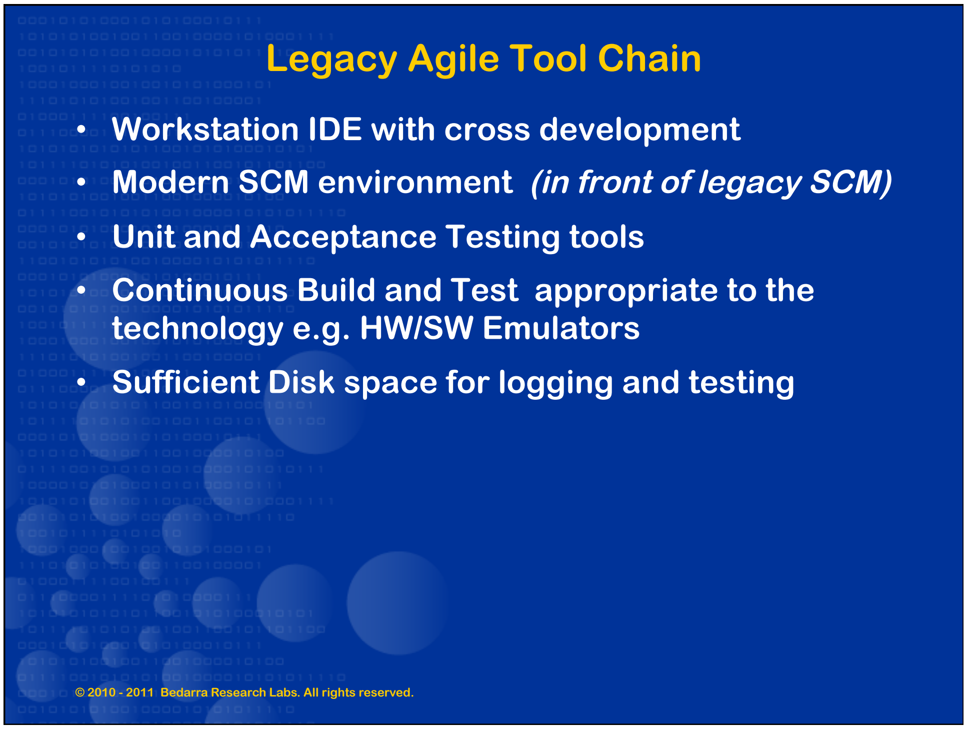
Legacy Archaeology

- Legacy Story Telling
- Listen to the dev and ops Data
- Look at the Code and Data Assets
- Look at the development value chain – people, process and tools
- Publish the Stories, Data and Value Chain and put the essence on the wall



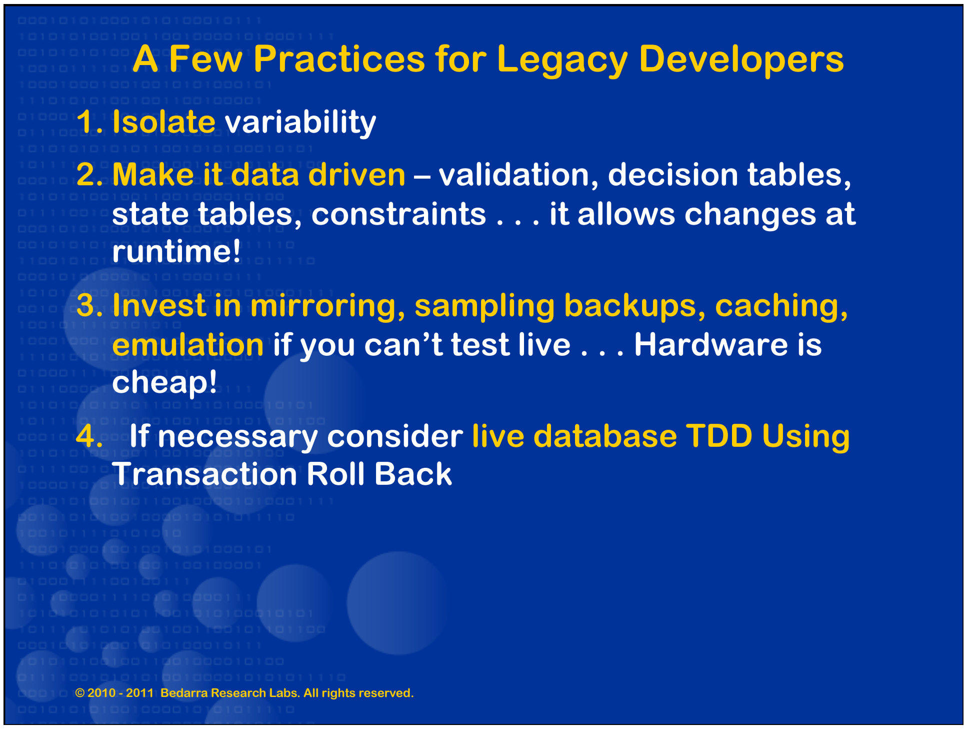
Legacy Agile Team

- appropriate skills for all areas
- cross train /pair Legacy and non-Legacy Developers
- Use playing Coaches who know the territory



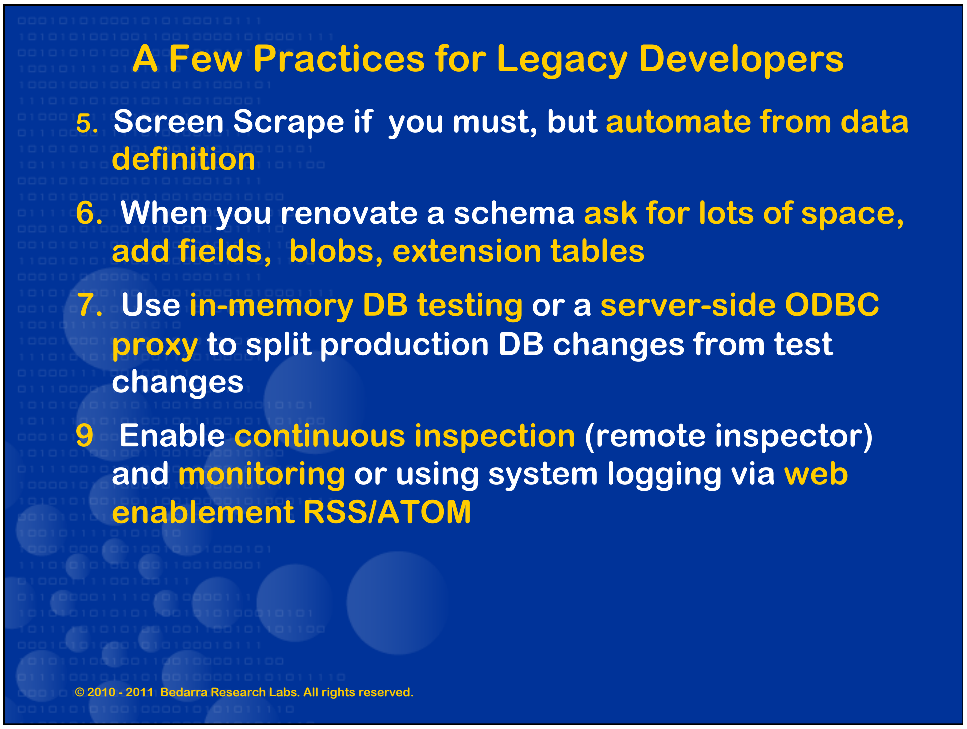
Legacy Agile Tool Chain

- Workstation IDE with cross development
- Modern SCM environment *(in front of legacy SCM)*
- Unit and Acceptance Testing tools
- Continuous Build and Test appropriate to the technology e.g. HW/SW Emulators
- Sufficient Disk space for logging and testing



A Few Practices for Legacy Developers

1. Isolate variability
2. Make it data driven – validation, decision tables, state tables, constraints . . . it allows changes at runtime!
3. Invest in mirroring, sampling backups, caching, emulation if you can't test live . . . Hardware is cheap!
4. If necessary consider live database TDD Using Transaction Roll Back

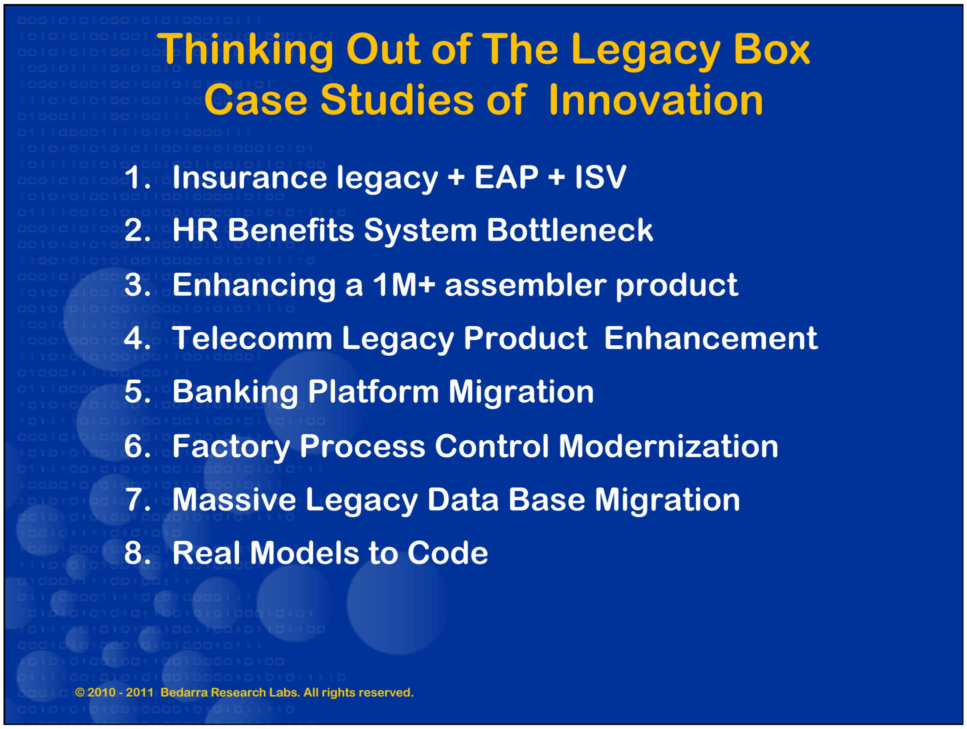


A Few Practices for Legacy Developers

5. Screen Scrape if you must, but automate from data definition
6. When you renovate a schema ask for lots of space, add fields, blobs, extension tables
7. Use in-memory DB testing or a server-side ODBC proxy to split production DB changes from test changes

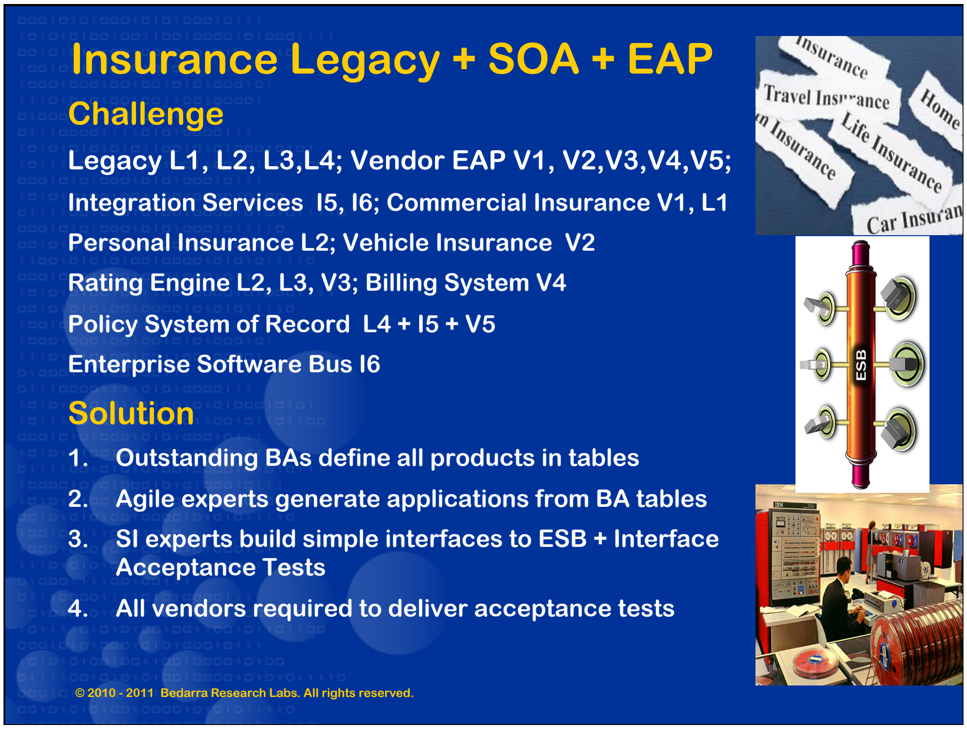
9 Enable continuous inspection (remote inspector) and monitoring or using system logging via web enablement RSS/ATOM

We need a ~~miracle~~ Innovation!



Thinking Out of The Legacy Box Case Studies of Innovation

1. Insurance legacy + EAP + ISV
2. HR Benefits System Bottleneck
3. Enhancing a 1M+ assembler product
4. Telecomm Legacy Product Enhancement
5. Banking Platform Migration
6. Factory Process Control Modernization
7. Massive Legacy Data Base Migration
8. Real Models to Code



Insurance
Travel Insurance
Home
Life Insurance
n Insurance
Car Insuran
ESB

Insurance Legacy + SOA + EAP

Challenge

Legacy L1, L2, L3,L4; Vendor EAP V1, V2,V3,V4,V5;

Integration Services I5, I6; Commercial Insurance V1, L1

Personal Insurance L2; Vehicle Insurance V2

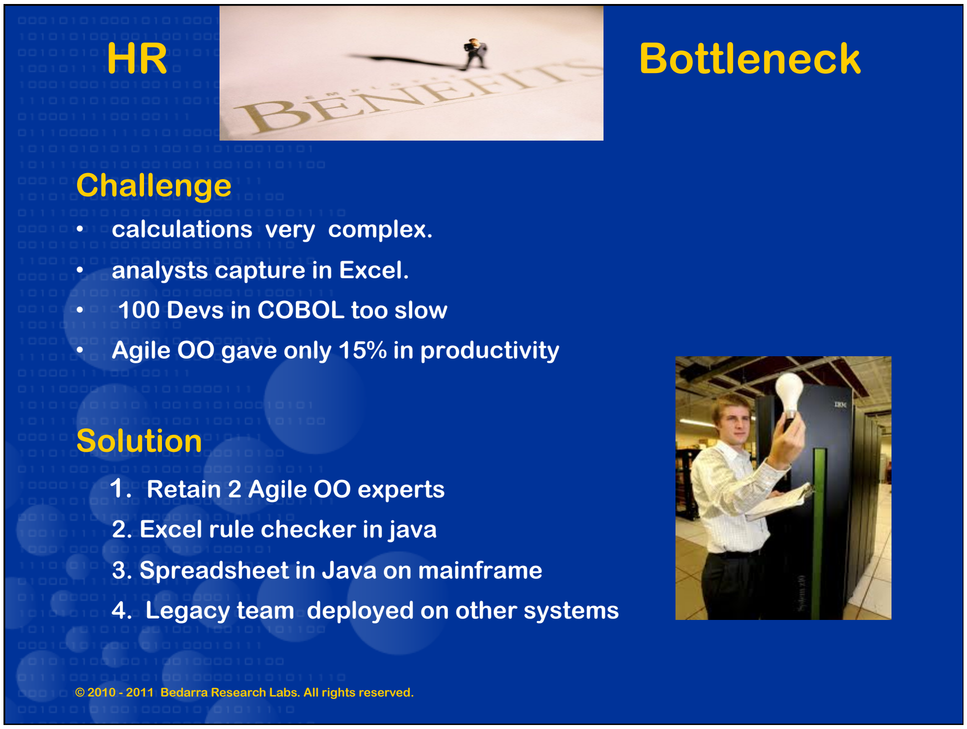
Rating Engine L2, L3, V3; Billing System V4

Policy System of Record L4 + I5 + V5

Enterprise Software Bus I6

Solution

1. Outstanding BAs define all products in tables
2. Agile experts generate applications from BA tables
3. SI experts build simple interfaces to ESB + Interface Acceptance Tests
4. All vendors required to deliver acceptance tests



HR Bottleneck

Challenge

- calculations very complex.
- analysts capture in Excel.
- 100 Devs in COBOL too slow
- Agile OO gave only 15% in productivity

Solution

1. Retain 2 Agile OO experts
2. Excel rule checker in java
3. Spreadsheet in Java on mainframe
4. Legacy team deployed on other systems

Enhance Mission Critical Assembler Application

Challenge

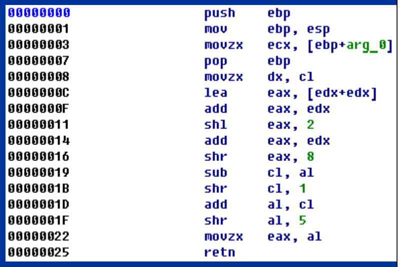
Enhance a 1M lines of assembler product

No legacy knowledge of code base

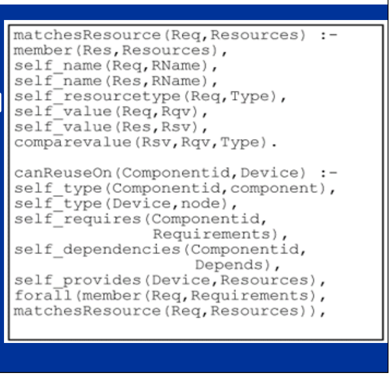
Solution

Translate assembler into Prolog rules

Use Prolog to answer dataflow, control flow questions to enable safe modification of the code



```
00000000                push    ebp
00000001                mov     ebp, esp
00000003                movzx   ecx, [ebp+arg_0]
00000007                pop     ebp
00000008                movzx   dx, cl
0000000C                lea     eax, [edx+edx]
0000000F                add     eax, edx
00000011                shl     eax, 2
00000014                add     eax, edx
00000016                shr     eax, 8
00000019                sub     cl, al
0000001B                shr     cl, 1
0000001D                add     al, cl
0000001F                shr     al, 5
00000022                movzx   eax, al
00000025                retn
```



```
matchesResource(Req,Resources) :-
member(Res,Resources),
self_name(Req,RName),
self_name(Res,RName),
self_resourcetype(Req,Type),
self_value(Req,Rqv),
self_value(Res,Rsv),
comparevalue(Rsv,Rqv,Type).

canReuseOn(Componentid,Device) :-
self_type(Componentid,component),
self_type(Device,node),
self_requires(Componentid,
                  Requirements),
self_dependencies(Componentid,
                        Depends),
self_provides(Device,Resources),
forall(member(Req,Requirements),
matchesResource(Req,Resources)),
```

Modernize Legacy Factory Automation

Challenge

$$$ investment in proprietary automation system

Must to integrate new systems and new GUI-based control rooms

Solution

1. Modify interface to control units using TCP/IP

1. Implement HTTP and ATOM

2. Client devices are Web 2.0 appliances or PCs

Enhancing a Legacy Telecomm HW/SW Product

Challenge

Successful Legacy Telecom product needs be upgraded

Product built in proprietary hardware, Os, languages …

3 Attempts to Greenfield replace have failed!

Solution

1. Move to VME bus , coprocessor, TCP/IP and DMA to legacy processor
2. New capabilities run on new processor
3. Eclipse tools + legacy unit + CI servers
4. DMA interface enables monitoring and debugging

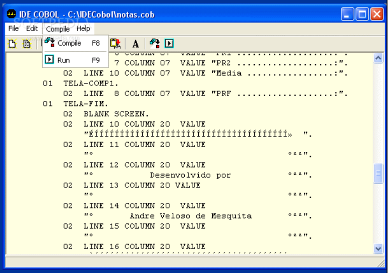
Migrate Banking Legacy Apps

Challenge

Legacy Hardware/Software Platform Too Costly to Maintain!
Rewrite estimate exceeds budget and has huge risk

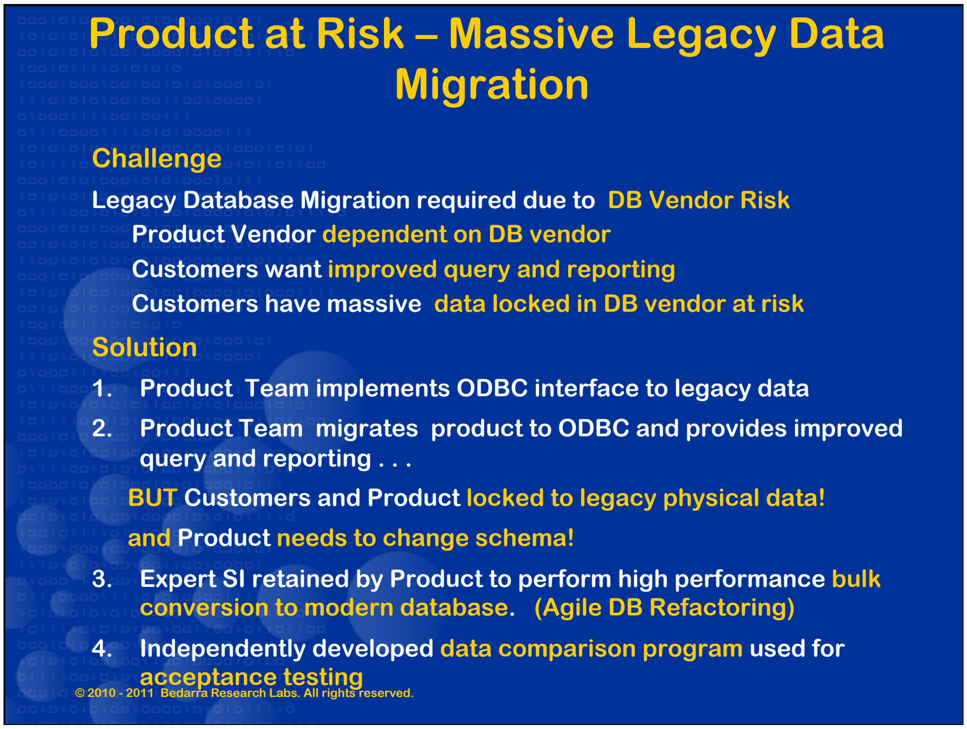
Solution

1. Expert ISV ports software platform to modern commodity platform. ISV validates using ATs derived from monitoring.
2. Expert SI Migrates banking apps ATs using regression testing and selected new acceptance tests
3. Upgrade modern IDE tooling



```
             7 COLUMN 07  VALUE "PR2 ...................:".
      02  LINE 10 COLUMN 07  VALUE "Media .................:".
  01  TELA-COMP1.
      02  LINE  8 COLUMN 07  VALUE "PRF ...................:".
  01  TELA-FIM.
      02  BLANK SCREEN.
      02  LINE 10 COLUMN 20  VALUE
          "ÉÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍÍ»  ".
      02  LINE 11 COLUMN 20  VALUE
          "º                                          º²²".
      02  LINE 12 COLUMN 20  VALUE
          "º            Desenvolvido por              º²²".
      02  LINE 13 COLUMN 20 VALUE
          "º                                          º²²".
      02  LINE 14 COLUMN 20  VALUE
          "º         Andre Veloso de Mesquita         º²²".
      02  LINE 15 COLUMN 20  VALUE
          "º                                          º²²".
      02  LINE 16 COLUMN 20  VALUE
```

Total Cost 10-20% of rewrite estimate in 25% of estimated time!



Product at Risk – Massive Legacy Data Migration

Challenge

Legacy Database Migration required due to DB Vendor Risk

Product Vendor dependent on DB vendor

Customers want improved query and reporting

Customers have massive data locked in DB vendor at risk

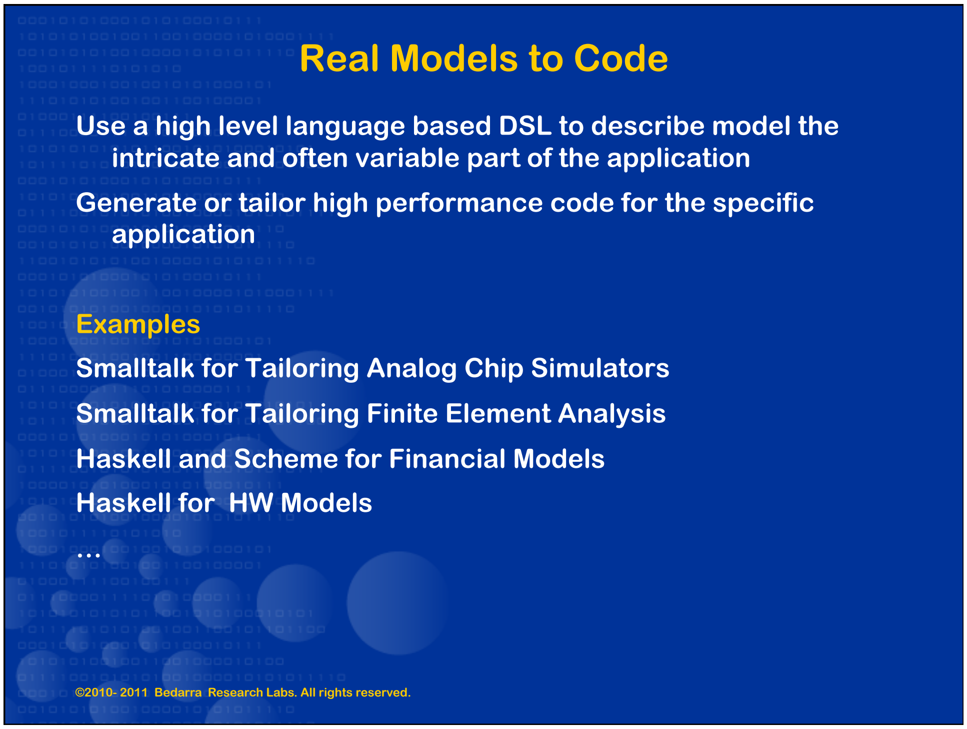
Solution

1. Product Team implements ODBC interface to legacy data
2. Product Team migrates product to ODBC and provides improved query and reporting . . .

BUT Customers and Product locked to legacy physical data!

and Product needs to change schema!

3. Expert SI retained by Product to perform high performance bulk conversion to modern database. (Agile DB Refactoring)
4. Independently developed data comparison program used for acceptance testing



Real Models to Code

Use a high level language based DSL to describe model the intricate and often variable part of the application

Generate or tailor high performance code for the specific application

Examples

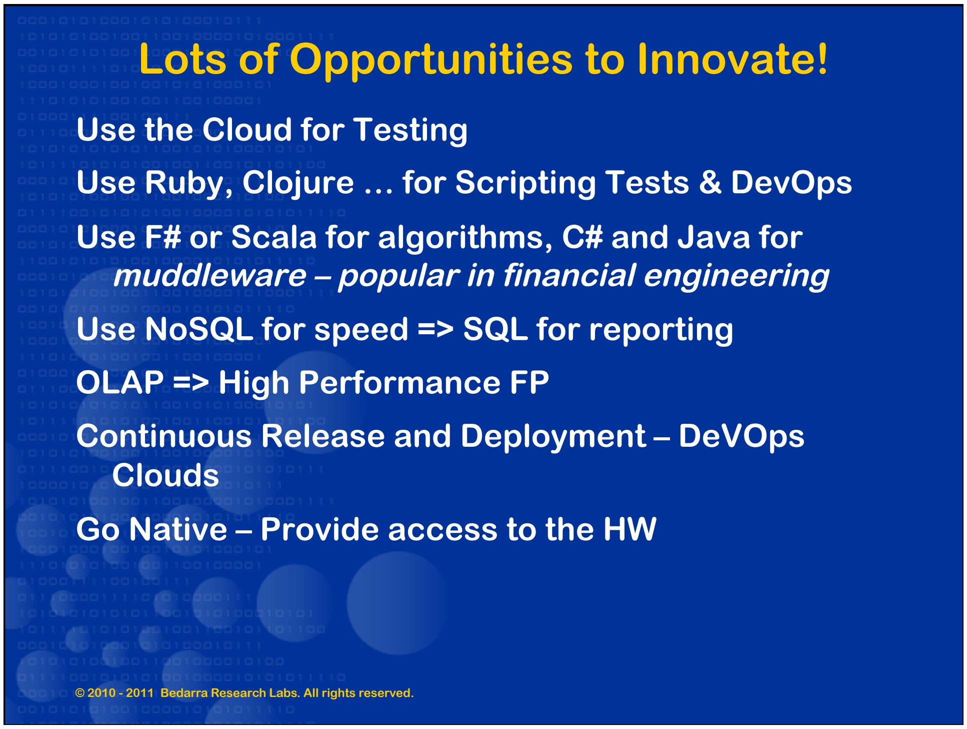
Smalltalk for Tailoring Analog Chip Simulators

Smalltalk for Tailoring Finite Element Analysis

Haskell and Scheme for Financial Models

Haskell for HW Models

…



Lots of Opportunities to Innovate!

Use the Cloud for Testing

Use Ruby, Clojure … for Scripting Tests & DevOps

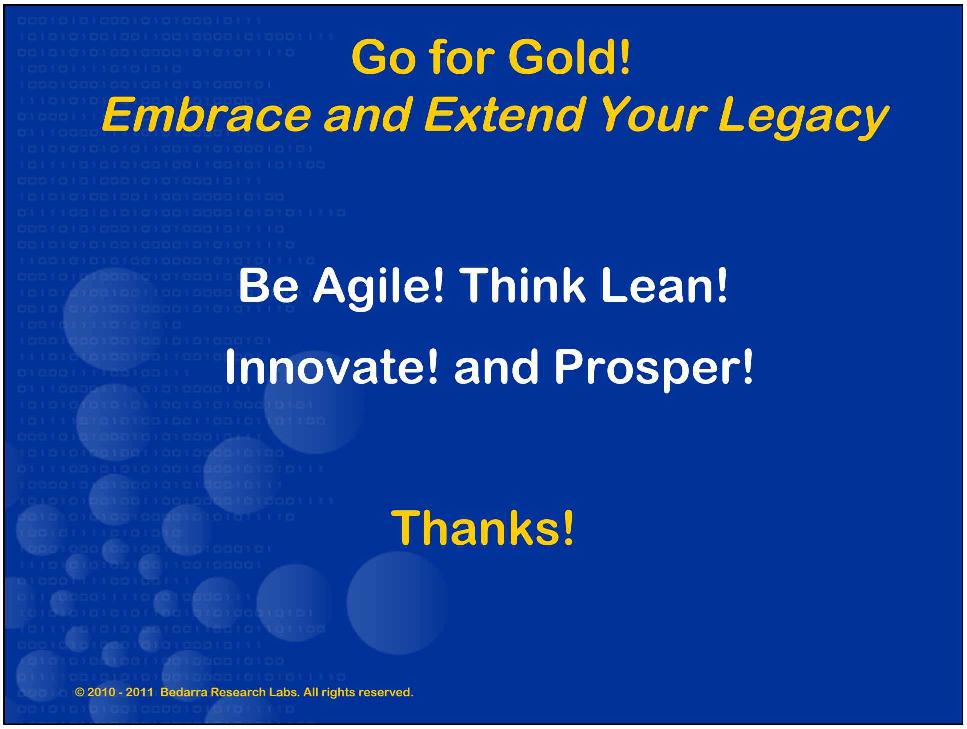
Use F# or Scala for algorithms, C# and Java for *muddleware – popular in financial engineering*

Use NoSQL for speed => SQL for reporting

OLAP => High Performance FP

Continuous Release and Deployment – DeVOps Clouds

Go Native – Provide access to the HW



Go for Gold!
Embrace and Extend Your Legacy

Be Agile! Think Lean!

Innovate! and Prosper!

Thanks!

CHICAGO

INTERNATIONAL
SOFTWARE DEVELOPMENT
CONFERENCE

goto;
conference

Conference: April 23-24, 2013 Training: April 25-26, 2013