



Refactoring and the other R: Why, When, and How!

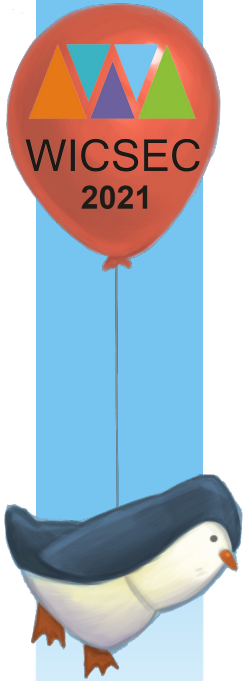
Curators: Jeremy Toulouse/Bart Mason

Menti Poll:

What Does Replatforming mean to you?

What Does Refactoring mean to you?

What is the Difference?



What is Re-factor?

- Re-factor:

- Techie – Convert existing *computer code written in one language to another development language* without changing its internal or external behavior.

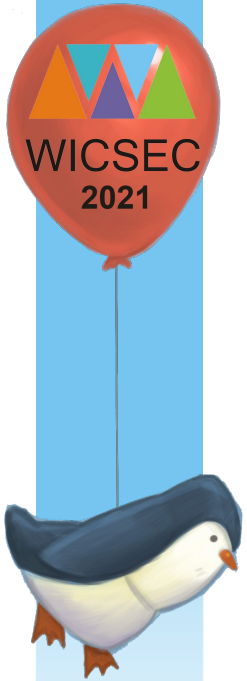
- Non-Techie – Translate from one language to another.

English

Hello

Japanese

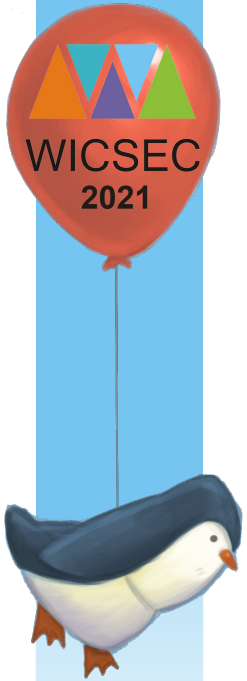
こんにちは



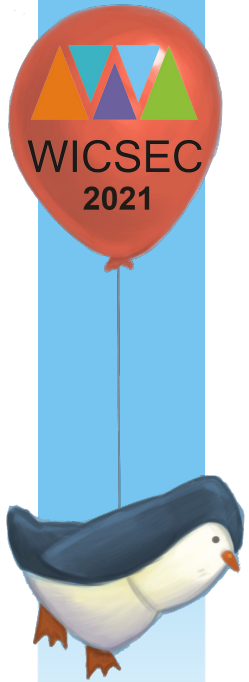
What is Re-Platform?

- Re-Platform:

- **Techie** – *Lifting the complete application from an on-premise environment to a new cloud hosted environment without changing its internal or external behavior. It is also referred to as a rehost strategy.*
- **Non-Techie** – Moving your house from one location to another.



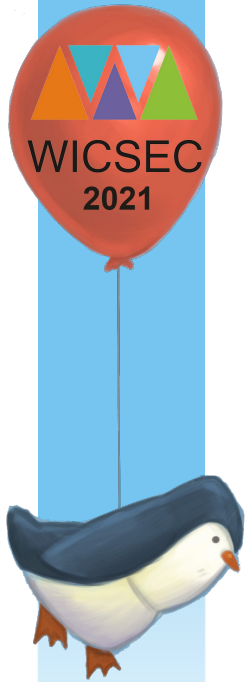
Exercise in Reality:



- You are a Child Support Director.
- You want to move your program in the direction that Child Support Modernization presents.
- Your program has worked hard over the last decade to improve Establishment performance and earn and save Federal Performance Incentives.
- Retirements have led to a lack of subject matter expertise, within both your program and your IT support program.
- New staff take a long time to onboard and some leave because of the complexity of the aged mainframe system and multiple sub-systems needed to support the 20+ year old business model.
- Staff don't have access to the data they need to make better decisions because several important interfaces were never automated.
- A new business model would be very hard to support because the mainframe only supports caseload management and the IT staff do not have the time or ability to develop more task-based functionality, which would support the program in overcoming the lack of expertise in some staff.
- Constant leadership changes have resulted in a revolving door of ideas, vision, and funding opportunities, with none of the above creating stability or a clear path forward.
- Many staff believe in the dead-beat dad punitive approach!

Problem Statement:

- Your Child Support Program has a developer and batch resource issue. Any system change request takes 12-18 months and flaws known only to developers that have retired, are leading to errors that are impacting both customer's lives and stretching an already thin budget.
- Your Child Support Program is the last application on the mainframe platform which is further stretching the already thin budget.
- Your Child Support Program has been notified that the mainframe platform will be decommissioned within a couple of years.
- Your Child Support Program is required to integrate into an enterprise level CLOUD platform.
- The current IV-D automated system replacement costs using next generations systems vary, but most solutions have an realistic total cost of over \$100 Million.



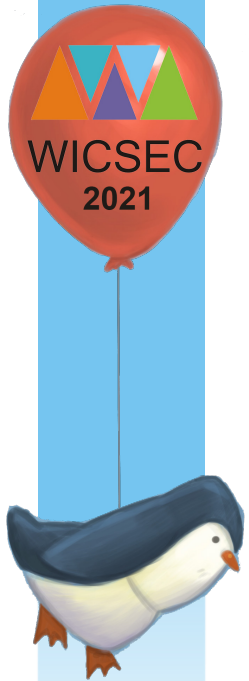
Replatforming Pros/Cons

■ Pros

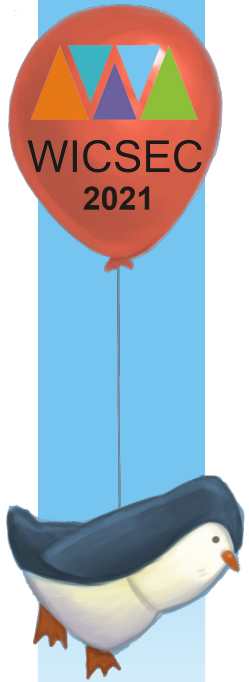
- Reduced annual maintenance and operations costs.
- Increased IT support capacity with additional development and other technical resources.
- Meets the needs, the impending state changes will create.
- Become a part of the enterprise.
- Take a step towards modernization

■ Cons

- There will be no immediate functional benefits!
- Securing several million dollars is complicated because many see Replatforming as only a temporary mitigation strategy.
- Most likely, your existing staff will be dedicated to this project, while also maintaining operations.
- Additional development and other technical resources will still be required and difficult decisions will have to be made.
- You will have to decide whether to migrate to the CLOUD or a client/server structure.
- Unidentified/unplanned costs.



Best Practices



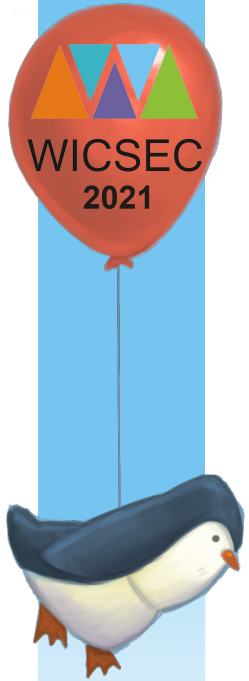
Business

- Focus on your Business Model! Be a sponsor for Change!
- Identify what your customers want!
- Identify the Organizational change necessary to be able to plan, manage, and implement projects! You need teams that understand the vision and have the expertise to get you there when solving problems.
- Identify the business process changes that will be more effective, efficient, and simpler to manage!
- Identify the cultural changes necessary to increase customer and individual staff changes and engagement!

Information Technology

- Focus on the child support business model, if you don't, you won't have program buy-in!
- Meet with other businesses (not just child support agencies) that have re-platformed.
- Meet with all vendors that do replatforming (this isn't new and there are important lessons to learn from other industries).
- Be transparent so that messaging shared between IT and the program and to, other stakeholders, is consistent and reliable. Doubt will always slow things down and derail projects!

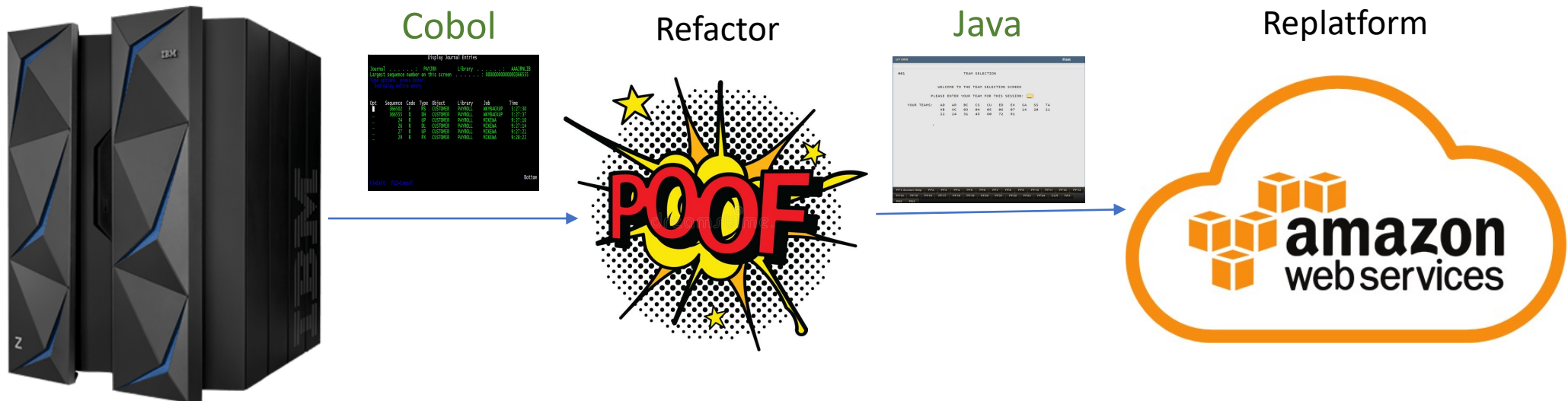
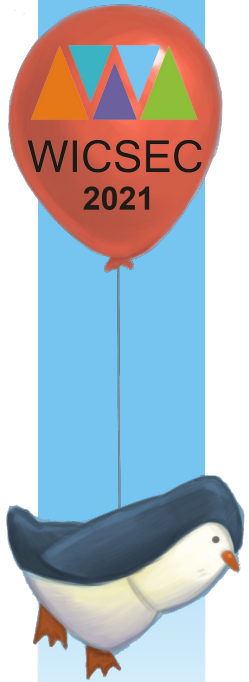
COBOL running on the Mainframe is bullet proof so why change?



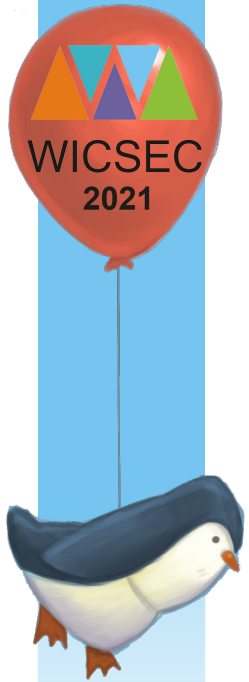
- The cost of running a Mainframe was over \$1,000,000 a year.
- Items such as licensing and storage were increasing in cost every year.
- Unable to do a full disaster recovery due to the data connection requirements.
- Finding COBOL developers was almost impossible.
- Relied on vendors to support the Mainframe
- Modern web applications were developed outside of the Mainframe due to language restraints.
- Incoming agents thought they were teleported back to 1959.

State of Utah Project Overview Simplified

- Re-factor the current Office of Recovery Services Information System (ORSIS) application written in COBOL to Java.
- Re-Platform our ORSIS application to Amazon Web Services (AWS).



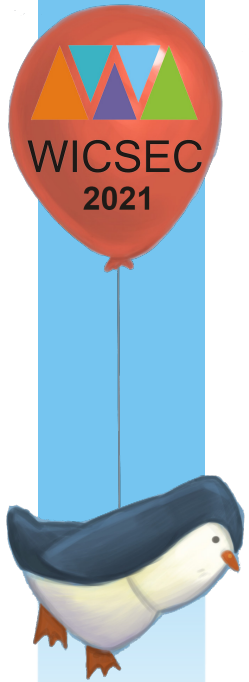
Current Mainframe Environment



- The State of Utah's Mainframe is an enterprise server that processes large, database-dependent applications. It also communicates with other systems both inside and outside the State of Utah.
- Mainframe located at the State's main data center: IBM Z13 rated at 406 MIPS with 120GB of memory. ORSIS separated into 4 LPAR's: Development, Initial Test, User Acceptance Test and Production.
- 2.5 Million lines of code written in COBOL.
- Approximately 500 DB2 (Database) tables.
- 900,000 transactions hitting the ORSIS database per day.
- 74,000,000 calls to the ORSIS Database per day
- Timeframe for nightly batches: 50,000,000 DB2 calls within a 4-hour window with the core batches completing before midnight.

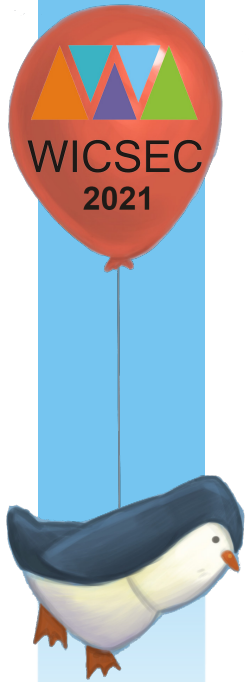
State of Utah Project Overview

- Translate current Cobol language to Java.
- Migrate IBM DB2 database from the Mainframe to a IBM DB2 LUW solution.
- Migrate the refactored Office Of Recovery Information System (ORSIS) from the state's Mainframe to a cloud infrastructure.
- The refactor and replatform project must be completed and in production within 18 months.



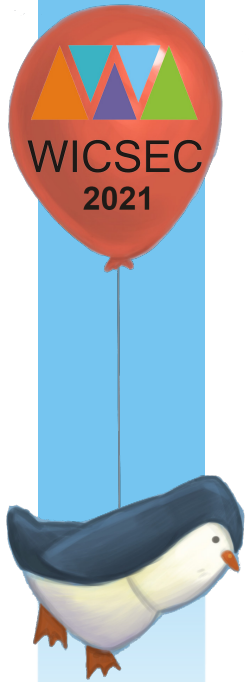
So how did we do?

- Our migration began in June of 2019.
 - The ORSIS code was refactored from COBOL to Java.
 - Migrated to EC2 instances within AWS Cloud Services.
 - Performance tested and User Acceptance tested completely.
 - And put into production October of 2020.
-
- Project was on time and on budget.

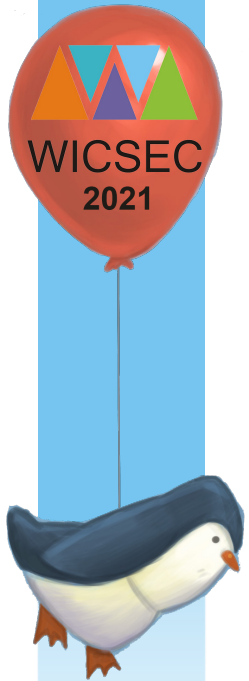


What did we learn over 18 months?

- IT staff struggles
- Hidden and unexpected costs
- OCSE and the IRS
- Interfaces
- Environments
- Dry Run Practices
- Pilot
- Project Management



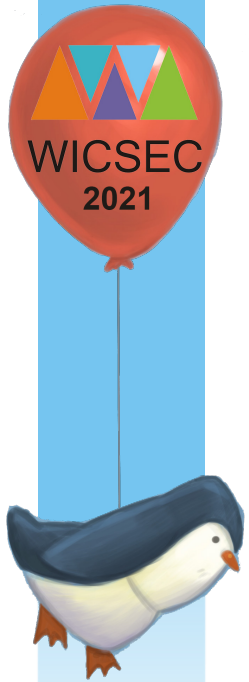
IT Staff Struggles



- The state's funding model for hosting support was not geared to give us the level of support required to provision the environment:
 - Used contractors to fill gaps.
 - Not enough Server Administrators to build systems.
- Underestimated the amount of work with this size of a project.
- It took much longer than expected to build cloud expertise:
 - Provisioning of cloud services
 - Developers went from developing code mainly in SCFPC to developing Java in Eclipse and other opensource tools.

Hidden and Unexpected Costs

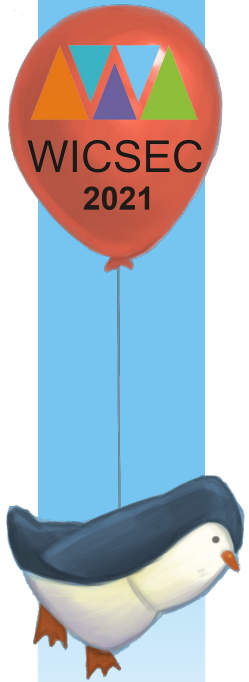
- Software licensing in most cases does not move from the Mainframe to the cloud.
 - IBM DB2 LUW - \$2.5 million for 3-year licensing
 - Spectrum UAM (address cleansing)
 - BMC Distributed Control M
 - IBM Data Studio for stored procedures
- Custom applications did not migrate.
- It took weeks to months to renegotiate license costs with vendors. Start the purchasing as soon as possible.



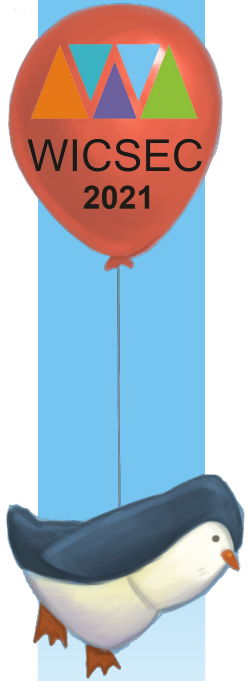
Hidden and Unexpected Costs Cont..

The cloud isn't one "thing". It's multiple "things" that are put together to make the cloud. Each "thing" comes with a cost.

- EC2
- EC2 Autoscaling groups
- VPC
- Inspector
- Cloudwatch
- Cloudwatch Logs
- Config
- Event Bridge
- SNS
- ELB
- S3
- SecurityHub
- GuardDuty



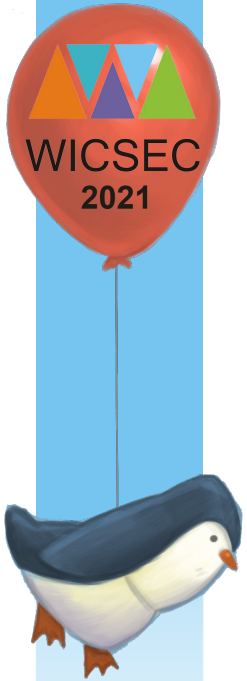
What the IRS requires..

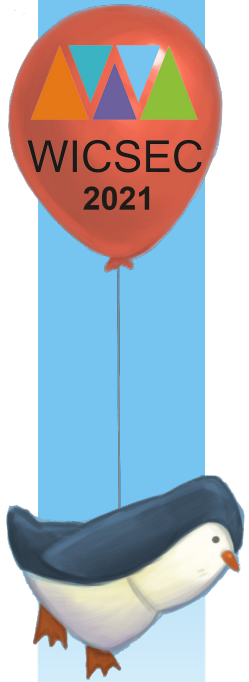


- The IRS requirements to move FTI to the cloud can be found at:
 - <https://www.irs.gov/privacy-disclosure/cloud-computing-environment>
- IRS 45-Day notices (PUB1075 pg. 139-140):
 - AWS
 - AWS contract provider
 - State's IT department
 - GovCloud
 - Code Migration Vendor
- Complete the requirements found on page 106-107 in PUB1075 (<https://www.irs.gov/pub/irs-pdf/p1075.pdf>).
- Submit the FTI Cloud Notification Form with the 45-Day notifications found at the bottom of <https://www.irs.gov/privacy-disclosure/cloud-computing-environment>.

What OCSE requires..

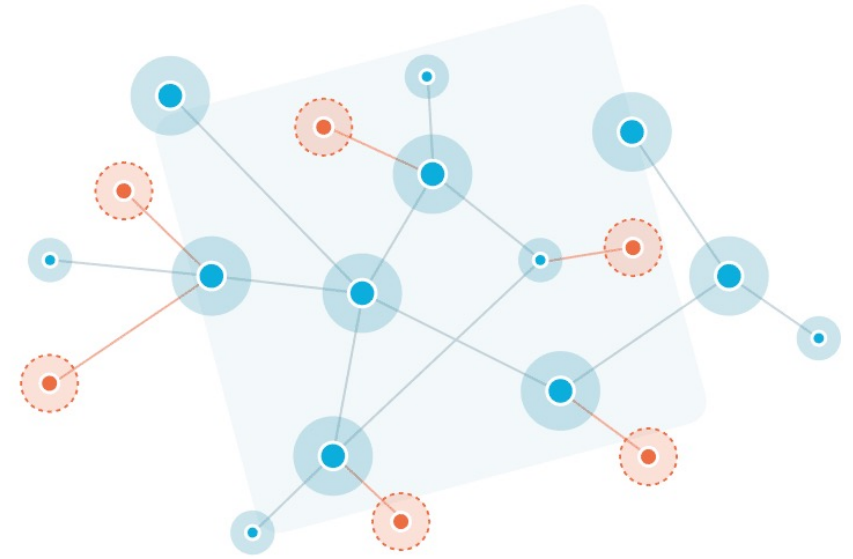
- OCSE does not allow production data in test environments.
- OCSE requires answers to the following questions before any type of data connection (non- production / production) is made to OCSE:
 1. The cloud solution must reside on a Federal Risk and Authorization Management Program (FedRAMP) compliant system. Is your chosen solution FedRAMP certified?
 2. Do you have a completed 3PAO report for the chosen Cloud solution? The cloud solution will usually provide this to demonstrate their security posture?
 3. Define what FedRAMP service(s) model you will be using SaaS/IaaS/PaaS/EaaS/XaaS?
 4. The State and Cloud Service Provider (CSP) should follow the data retention policies provided in the OCSE Agreement. Please describe how these policies will be maintained?
 5. The data used by e-IWO should ONLY be used for the purposes it was intended for. Please describe what measures will be in place to accomplish this?
 6. Provide a network and data map of the new system.





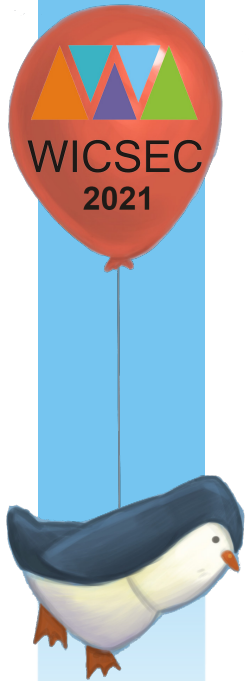
Interfaces

- ORSIS had over 100 external interfaces with different interface partners.
- Many of the interfaces just needed to be repointed to a new sFTP server in AWS.
- Some interface partners needed to make coding changes to connect to our new system.
- **Difficulty in getting interface partners to respond in a timely manner.**
- Not a high priority to some partners.
- One data connection was not setup until the day before Pilot testing.

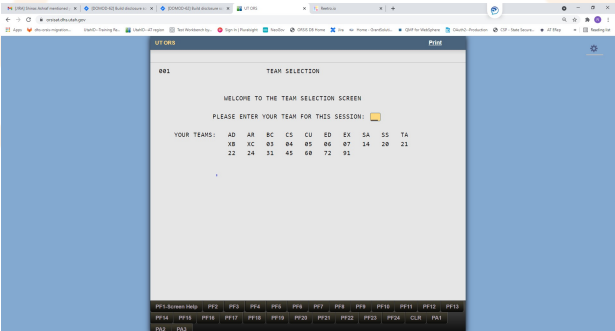


Dry run practices

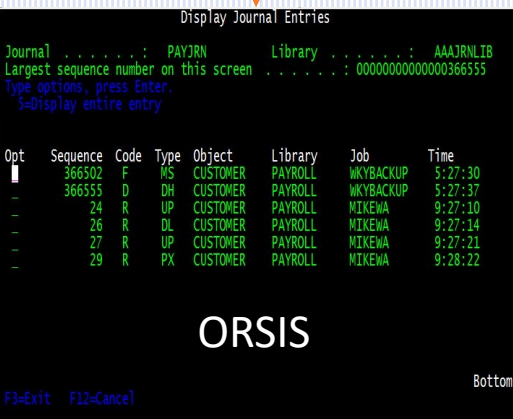
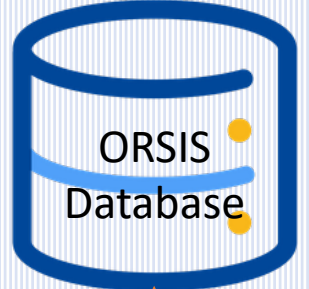
- Dry run practices are walking through each step of the migration to production process.
- Dry runs started on a Friday night and run through Sunday.
- The business staff were involved including the Director.
- Practice each step of your migration to production to find issues such as how long a backup would take.
- Each step needs to be performed except for going live.
- Thanks to several paper go lives and four practice runs going live was almost anticlimactic.



Pilot phase overview..



Re-factored ORSIS



ORSIS

State Datacenter



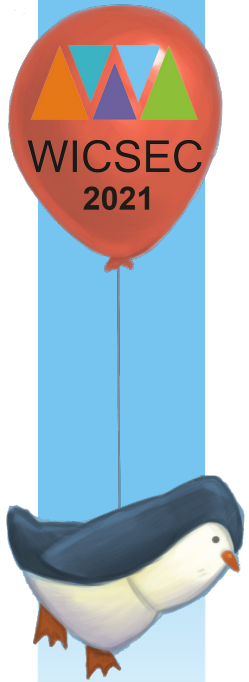
Pilot Users



Users

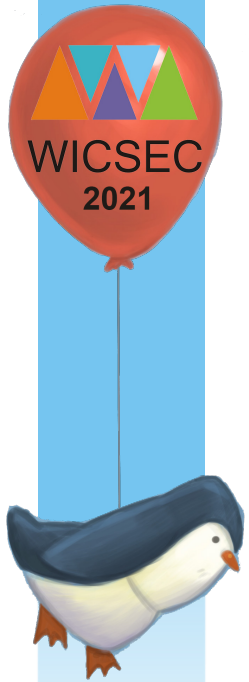
Pilot phase results

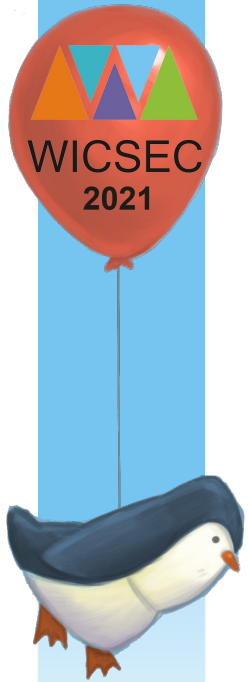
- The pilot phase ran over a 4-week timeframe.
- Pilot users would spend 4 hours each day in the new system and 4 hours in the old system.
- 20 users would be added to the pilot phase each week.
- One major issue with a couple of screens was found during pilot that was not found in User Acceptance Testing.
- The issue was resolved quickly, and no other major issues were found.
- Pilot went so well that at the end of the second week we decided to bring all 80 Pilot users into the pilot phase rather than 20 at a time over the next 2 weeks.



Project Management

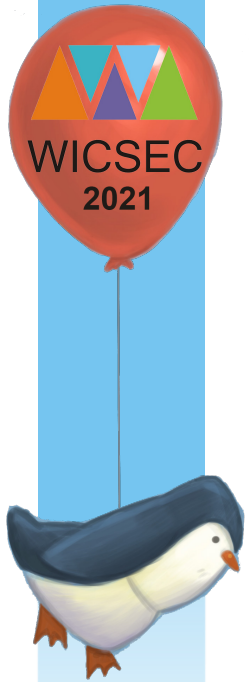
- Project Management will make or break your project.
- Daily 15-minute standup meetings with the project groups made a big difference in getting the schedule caught up.
- Weekly project schedule meetings.
- Weekly project status and risks meetings.
- Monthly project status meeting with business stake holders.
- Constant communication and involvement of all the staff members working on the project.





Things that did go well

- AWS environment was well documented
- It's easy to allocate and provision new resources. It's just as easy to deprovision resources.
- AWS environment performed exactly as the documentation described.
- Thanks to several paper go lives and four dry run practice runs, going live was almost anticlimactic.
- AWS made resources available to help answer technical questions.
- Vendor (Deloitte) was very easy to work with.
- Deloitte had strong project management involvement from the beginning
- Resulting migrated code was fast with pleasantly few defects.
- Additional change request purchased to help provide knowledge transfer and mentoring in the new environment went smoothly.
- Multiple layers of testing with external partners, unit testing, UAT and performance testing helped ensure delivered code performed well.
- User Acceptance Testing was completed in 6 weeks on the entire system.
- The conversion model assured that ORS could continue to make changes to the old business system until several months prior to deployment.



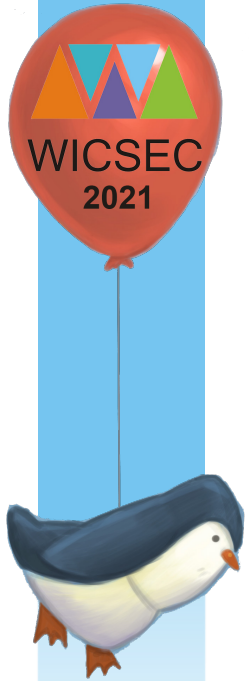
Research

- Go to conferences:

- WICSEC
- Cloud Vendors:
 - Google
 - Google Cloud Events
 - Google I/O
 - Amazon
 - AWS re:Invent
 - AWS Public Sector Summit
 - Microsoft
 - Azure summit

- Cloud providers provide free resources to learn about their products.
- Talk to other states.

Thank You!



Questions?

