FINAL - "Low Code/COTS" Vendor Presentations

NCCSD Systems/Data Analysis Workgroup
Systems Modernization Vendor Webinars Oct. 18/Oct. 25/Nov. 1, 2019
"Replatforming/Refactoring"(R/R) and "Low Code/COTS" (LC/COTS) Approaches
Recap and Most Interesting Lessons/Similarities from the Vendors

Caution: This document contains information presented at confidential sessions between specific vendors and state representatives and may contain information that could be proprietary. Please consider this when sharing with non-State or your vendor staff.

9 Participating Vendors (in alphabetical order):

Accenture, Cambria Solutions, Cardinality, Conduent, Deloitte, IBM, Microsoft Dynamics, Protech, Redmane.

Terminology/Definitions:
This approach is so new in the child support community that terminology to describe it and its attributes is definitely different depending on the vendor, and seems to be evolving. A variety of terms used by the vendors in their presentations included:

- "Platform" solution, including the concept of an "All-In Platform" vs. "Platform Integration" – building almost everything into the platform vs. using the platform in conjunction with other components to create the full child support system.
- Configuration – "clicks not code", "point and click configuration", "drag and drop"
- "Accelerators"
- "No code" – meaning there is no coding done. Everything is configured.
- RAD (Rapid Application Development) features = low code or no code development
- "Purpose Built" or "Custom Built" Platforms – platforms created for the specific purpose of managing human services or even child support specific program services.
- Platform as a Service (PaaS) – subscription based platform
- COTS = Purchased as Products, but Platforms = Purchased as Services
- COTS = "shrink wrapped," "out-of-the-box" (or "native") functionality
- 3rd Party or "Peripheral" Products or Tools (such as document generation and/or content management tools, separate "rules engines," etc.)
- Platform vendors (those vendors that provide a platform on which to create a CS application) vs. System Integrator vendors (those vendors that work with the platform to create a CS application)

Approaches:
Again, this seems to be a rapidly changing area but two major approaches could be seen:

- Taking existing platforms built for other areas (private sector, child welfare) and using System Integrator vendors to tweak them to fit child support. Examples were CRM (Customer Relationship Management) or ERP (Enterprise Resource Planning*) platforms; or
- Using a "Purpose Built"/"Custom Built" Platform that is developed specifically for human services and/or child support.

In the webinars:

- Accenture and Deloitte discussed/showed us solutions based on a Salesforce CRM platform

- Cambria Solutions, IBM, and Microsoft Dynamics discussed/showed us solutions based on a Microsoft Dynamics CRM/ERP platform
- Cardinality showed their open source "purpose built" platform for human services
- Redmane discussed/showed a combination approach: a "purpose built" platform for the majority of child support functions combined with an Acumatica Financial platform tweaked for child support financials.
- Protech made the argument that their existing solution was "low code" because it theoretically is easy to transfer between states, even though it did not appear to be based on a platform like the others.
- Conduent did not seem to lean to one platform or another.

Vendors have different takes on whether it makes sense to:
- Take various platform components and combine them, or
- Use multiple 3rd party/peripheral products in combination with the platforms, or
- Go "all in" and build everything into one platform with only a very few 3rd party/peripheral additional products.

Major takeaways and consistent themes from the sessions:

- *No one* has implemented a child support system low code platform solution yet. Indiana has just finalized procurement, and may be the first to attempt. Since the low code platforms have been implemented for child welfare, many of the vendor comments seemed to be predicated on how the approach worked there.

- This approach is supposed to save time on development and testing by:
  - Platforms start with many common services already "built in", e.g. security, reporting, audit logging, error handling, workflow, and often have others such as content management, document generation, mobile awareness or readiness.
  - Much of the discussion centered on the "configuration" or "accelerator" concept. Terminology used: "clicks not code", "point and click configuration", "drag and drop", etc. Changing things like the screen 'look and feel' or labels on fields is supposed to be easy. One vendor stated: "The built-in widgets, tools and configurable screens allow development of many child support functions in a short amount of time." Another stated "Low code/COTS solutions support standard case management functions including account, participant and workflow management. These capabilities can be integrated with core functions much faster with less errors thereby also reducing testing time".

- All of the vendors warned that the less customization needed, the better.
  - Estimates for using the out of the box functionality ranged from 50-70%. Custom code is distinguished from configurability in how much it strays from the basic platform. The level impacts release management, timeframe to implement, and cost. Different vendors had different examples of the areas that would need "custom" code. One stated "One area where customization will be required, however, is integrations (interfaces to share data with other systems such as FPLS as well as state systems)," but others pointed to the use of "pre-built connectors/APIs" to avoid custom code for interfaces.
  - Vendors also disagreed on how much of child support financials would need custom coding. One said "a significant portion" while others said that there can be "pre-built financial models specifically built for child support" that then use configuration to meet a state's specific requirements. Note that no one has yet built financials specifically addressing child support requirements.
  - Vendors brought up multiple times that if a state can be flexible about how its child support processes work, then it is more likely to be able to "leverage out of the box" functionality (i.e. fully leveraging the capability of new technology). If this is the case, the low code platform approach may be a good match and less costly. (It was interesting that this discussion sounds similar to more recent trends

with transfer systems – i.e. changing many of a state's business practices/requirements to meet the incoming transfer system's functionality in order to save time and money.)

- More work is supposed to be accomplished with less and/or different staff:
  - Platform vendors are doing much of the work to maintain/improve the base platform, and send out releases to their customers, so fewer or different types of IT staff at the states may be needed.
  - One vendor stated that this approach "opens the developer playing field to a much wider audience than the limited population of highly specialized programmers."
  - With configuration, fewer developers are needed to generate the same results.
  - Some vendors stated that more work can be done by business staff (e.g. creating new forms and reports) without specialized IT staff intervention.

- This approach seems to assume Cloud utilization via Platform as a Service (PaaS). One vendor stated it would be "…appropriate for agencies who want to get out of the hardware business, leverage an external party (provider) to manage and maintain the infrastructure, and eliminate their footprint in State Data Centers."

- Most if not all vendors said that the preferred project management methodology is Agile (in one form or another).

Timeframes and Cost.

For those vendors who answered (and many did not) the consensus seems to be that the cost should be less than a custom approach, and also possibly less than a transfer approach. This was also reflected in the length of time they stated for development and testing.

- All agreed on timeframe estimates of 2-3 years, although some said that timeframe was only for development, not including statewide rollout. Four years for everything *seemed* to be the outside estimate.
- Cost projections were $50M+; $10M-$40M; $40M-$70M; $40M-$100M with the caveats noted below. These estimates seemed to be for development and testing, not the other aspects of the project, and not total cost of ownership.
- The vendors noted that their cost and timeframe estimates could be significantly impacted by multiple factors, including state-specific requirements, level of customization, data conversion effort, number of deliverables, constraints, development approach (agile, waterfall or hybrid waterfall/agile), implementation approach, etc.
- Several vendors noted that states should pay attention to the cost structure with platforms (cost per user fluctuations, costs associated with not only the number of additional 3rd party/peripheral products but the complexity of maintaining the overall integration.) One vendor stated "The cost and support model for these types of systems are different than a traditional system transfer. The State needs to understand how the costs will be different (user licensing, type of programming resources, warranty, technical support) and how it will affect the program's costs once deployed, as well as the initial cost of construction."
- Vendors noted that data migration complexity and costs can be significant.
- Vendors also noted that maintenance/operations costs after implementation should be proportionate to the "build" cost, *but* depends on level of customization.
- One vendor said leveraging a low-code platform would, compared to a "custom build," "reduce the development time and cost by 25-30%." (Note that this vendor was only referencing *development*, not total cost of ownership.)

It is important to remember that no state has implemented this low code approach in child support so the above answers must be taken as best guesses from the vendors.

Miscellaneous:

- Steps to Prepare: Most vendors included the following as important items:
  - Evaluation of the current system, and Business & Technical Assessments are highly suggested.
  - Cleaning up any known data issues in preparation for data conversion.
  - Establishing a common enterprise vision, organization goals, long-term plan or roadmap, etc. up front.

- Interfaces and integration: Can be done with "APIs" (Application Programming Interface) or the platform itself might have "pre-built connectors". One vendor stated that this "facilitates integration – the modules can talk to the platform in any language and the platform is responsible for translating a request from one module to another in a form that the receiving module can understand and respond to."

- Continuous Improvement:
  - One vendor stated "As technology advances and the platform vendor enhances the platform to take advantage of the new advances, it is easy to upgrade the platform without disturbing the functionality built on top of the platform….the cost per user typically remains the same."
  - Another stated that "The products offer agencies the ability to offload most of the technology refresh of the child support system, and a continuous improvement in configurable tools to build and manage the system. The platforms will keep the agency current with security needs as well as technology products."
  - However, caveats were typically added that states will probably still have to bear the cost of upgrading any custom coding (e.g. for things like financial management) included in the solution.

- Use of Rules Engines in conjunction with the platforms: Often known as "Business Rules Management Systems" (BRMS). Examples given were InRule, Corticon, Camunda (open source), IBM ODM, and Oracle OPA. Vendor comments on using these: only use for very complex business logic that the platforms' out-of-the-box rules and workflow logic can't handle. Some vendors said they didn't think Rules Engines justified their cost.

- Workflow: Most platforms offer out of the box workflow – case routing and case assignment -- capability. Can place work in "queues".

- Dashboards: Platforms offer easily configurable and multiple dashboard views.

- Worker personalization: Workers can often personalize their "home page", dashboards, etc.

- "360° View": Most of the platforms offer the ability to visually see relationships between the various players in cases.

- Enterprise Solutions: Platform solutions can make sense if a state wants to use the same platform across multiple entities. May provide economies of scale for licenses, etc.

**Definition of ERP: Systems and software packages used to manage day-to-day business. Comes from the private sector, and typically is a "suite" of software that helps with accounting, procurement, supply chain, performance management, etc. and enables the flow of data between the business processes. Uses a common database and has a consistent look and feel. (See Wikipedia; "What is ERP?" at Oracle website; "What is ERP and why do you need it?" at Microsoft Dynamics website)*