

Customer Connected Engineering at patterns & practices [Draft]

Engaging Customers Throughout the Life Cycle!

J.D. Meier, Per Vonge Nielsen,
Ade Miller, Javed Sikander,
Tom Hollander, John deVadoss



patterns & practices

Proven practices for predictable results

From this ...

“Build it and they will come” ... NOT!

To ...

“Built with customers, for customers”



Agenda

1. Overview of Customer Connected Engineering (CCE)
2. CCE in the Product Life Cycle
3. Key CCE Activities
4. Why CCE
5. Customer Advisory Board Selection
6. Stories and Scenarios
7. Prioritization
8. 11 Guiding Principles for CCE



What is Customer Connected Engineering?

- **Customer Connected Engineering (CCE)** - Engaging customers during the planning, development, and release of deliverables.



Overview of CCE at patterns & practices

- We engage customers early and throughout the process.
- Customer Advisory Board influences what we ship.
- Customers help us ship better products that meet their needs.



Customer Connected Engineering at a Glance

	Core	Customer Connected Engineering
Exploration	<ul style="list-style-type: none"> • Go / No Go • Business Case • Product Backlog • Release Planning • Team / Role assignments • Vision / Scope 	<ul style="list-style-type: none"> • Broad Customer Survey • Customer Advisory Board Setup • Stories / Scenarios • Prioritization
Iteration 0	<ul style="list-style-type: none"> • Clarification of process • Clarification of responsibilities • Clarification of roles • Infrastructure Setup 	
Iteration N	<ul style="list-style-type: none"> • Iteration Planning • Daily Stand-Up • Mid-Iteration Checkpoint • Review • Retrospective • Internal Release (Optional) • Customer Release (Optional) 	<ul style="list-style-type: none"> • Stories / Scenarios • Prioritization • Demos • Product Drops • Feedback
Stabilization (Optional)	<ul style="list-style-type: none"> • Remaining Work Completed • Outstanding Bugs Resolved 	
Release (Optional)	<ul style="list-style-type: none"> • Documentation Updates • Incomplete Stories Removed • Final Test • Remaining Bugs Resolved • Release Bar Met 	<ul style="list-style-type: none"> • Feedback



Key CCE Activities

- **Customer Advisory Board.** The Customer Advisory Board is a set of customers that act as a sounding board for the project. This is a smaller set of customers that act as a proxy for the rest of your customer base.
- **Stories / Scenarios.** Customers share stories and scenarios. Stories and scenarios are narratives that capture and share usage scenarios for your product. The scenarios help show requirements in context.
- **Prioritization.** Customers help prioritize by providing input for the product backlog, the sprint backlogs and iteration planning sessions. The advisory board provides their prioritization ongoing. For some projects, we will open up a survey to the broader community to help with prioritization in the earlier phases before Vision / Scope.
- **Feedback.** Customers provide feedback during iterations and for release. What's important is that it's earlier instead of later. This helps course correct midstream instead of miss the mark at the end of the project.



Why CCE

- **Relevant scenarios.** Customers help you identify relevant scenarios based on their real problems they care about.
- **Prioritize, rationalize and refine.** Customers help you verify, prioritize, rationalize and refine the scenarios.
- **Feedback shapes design.** Customers can evaluate and provide feedback of your deliverables against their scenarios.
- **Raving fans.** Involvement helps build trust and increases probability for adoption and usage.
- **Transparency.** Customers better understand your trade-offs and have more visibility into your process.

Success depends on how engaged your Customer Advisory Board is and how representative they are of your target customer base!



Customer Advisory Board Selection

- **Key Contributors.** If you have found a reference solution that addresses some of the core challenges, consider engaging key contributors for the board.
- **System Integrators.** System Integrators can be aggregators of requirements for multiple customers, and watch out that they are still representative of the main stream.
- **ISVs.** Check that they are representative of the main stream. ISVs may address extensions compared to main stream customers, which may still be valuable.
- **MVPs.** MVPs are another source, and they certainly understand the directions. Often they are early adopters themselves and work with early adopter customers.

* This is in the context of the patterns & practices team at Microsoft.



Stories and Scenarios with CCE

- **Don't miss key scenarios.** A lot of software projects fail because they miss the scenarios.
- **Get stories direct from customers.** It's one thing to imagine or dream up scenarios, it's another to get them directly from customers.
- **Features don't equal scenarios.** A lot of working features don't necessarily aggregate up into working scenarios, or even the right scenarios.
- **Measure value by the problems solved.** The value of your deliverable can be measured by the problems it solves.
- **Evaluate against scenarios.** Ultimately, you can evaluate your deliverable against usage scenarios.



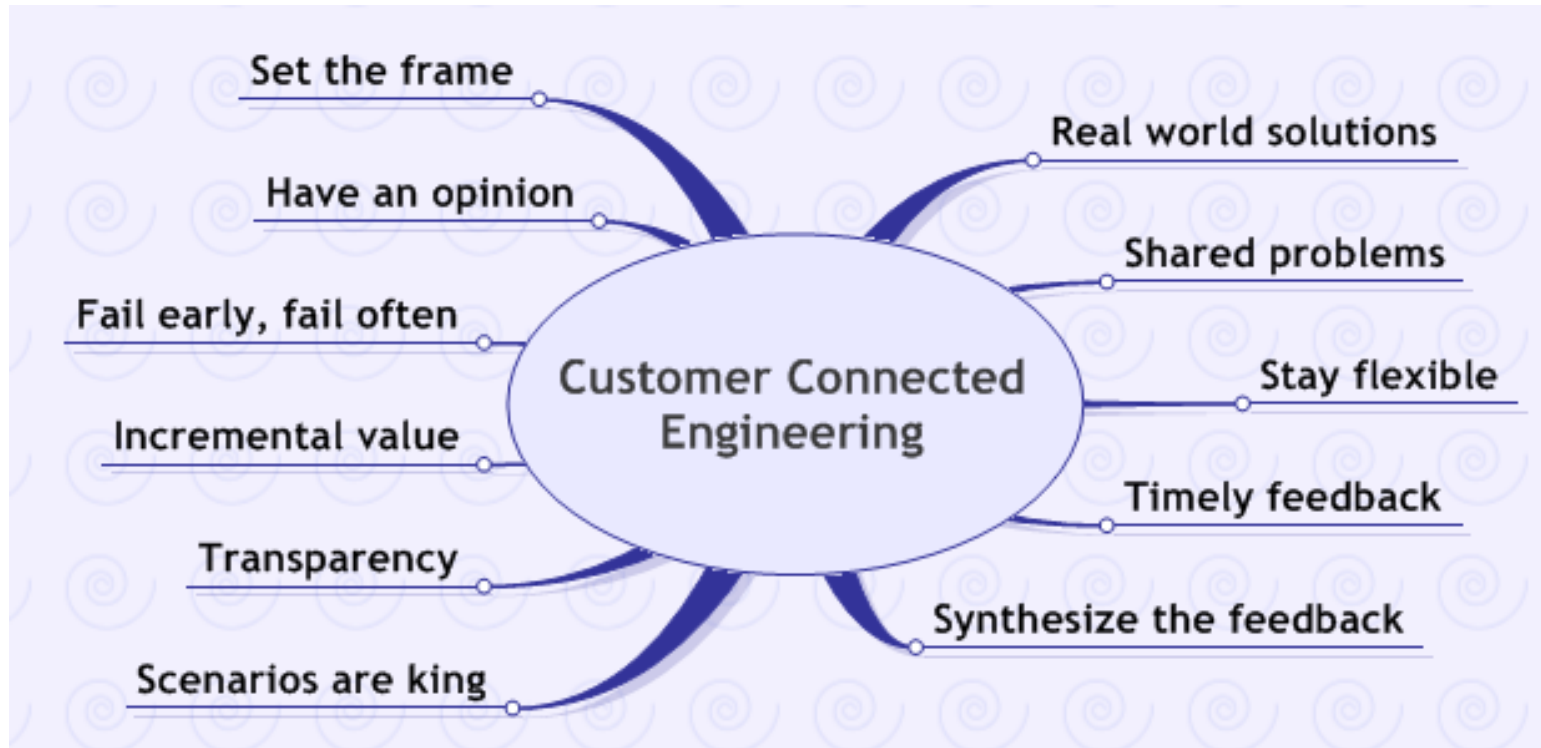
Guiding Principles

patterns & practices

proven practices for predictable results



11 Guiding Principles



Principle #1 - Set the Frame

- A frame is how you look at things.
- One frame could be an architectural overview.
- Another frame can be your product backlog.
- You need to frame the discussion and create something that people can react to.
- The more thoughtful the frame, the higher the quality feedback you get.
- You create the frame by figuring out the customers, their needs, and the business goals.
- You use the frame to help focus feedback and dialogue.



Principle # 2 – Shared Problems

- The customers you select for the Customer Advisory Board need to have first-hand experience with the problem.
- They need to care and be involved in the solution.



Principle # 3 - Have an Opinion

- **It's a piece of the pie.** Your opinion is a piece of the overall pie.
- **Get the input.** Your opinion is a piece of the pie, not the whole pie. Get the input from your customer advisory board to complete the pie.
- **Avoid “built by committee.”** Rationalize the input. Have an opinion so you can rationalize and understand the input



You don't want the doctor who can't take input and you don't want the doctor, who has no clue.



Principle # 4 – Synthesize the Feedback

- Step back and look across the scenarios and requirements.
- Look for common denominators.
- Prioritize across your highest ROI items.



Principle # 5 – Scenarios are King

- Scenarios are the backbone of Customer Connected Engineering.
- The end-to-end scenarios are one of the most important outcomes.
- Walk through stories and scenarios with customers.
- Customers can share their goals and their stories in detail.
- Have a set of straw man scenarios, before you engage with the advisory board.



Principle # 6 - Transparency

- Transparency is letting customers see inside your process to understand how things work.
- It's sharing your decision making approach so that customers understand how trade-offs are made.
- It's also about sharing design goals as you know them.
- It's also about making customers aware of important changes along the way, instead of at the very end when you ship.
- It's opening up the door to the workshop and letting customers watch and participate as you build your deliverables.
- When they understand why you made a decision / tradeoff, you are more likely to have a satisfied customer, even if they disagreed with a specific decision.



Principle # 7 – Incremental Value

- Find a way to flow value.
- As the project progresses, customers should get a sense that you are delivering value along the way.



Principle # 8 – Fail Early, Fail Often

- Share releases with your customers so they can share feedback.
- You don't want to be surprised when you're ready to ship. Share early and share often.
- Use the feedback to improve.



Principle # 9 – Timely Feedback

- A big benefit of Customer Connected Engineering is timely feedback.



Principle # 10 – Stay Flexible

- Be responsive to feedback.
- Acting on feedback will show customers you value their input and that it makes a difference.
- The more they see the impact, the more they'll engage.



Principle # 11 – Real World Solutions

- If you have a working implementation, you have a significant starting point.
- Where you can, find examples of specific customer solutions that solve some of the same scenarios and challenge you are facing.
- To speed up your success, rather than chase your competition, you can look to working solutions.



Are You Doing CCE?

patterns & practices

proven practices for predictable results



From This ... To That

From This To That
You dream up scenarios	Customers supply scenarios
Customer feedback after the fact	Customer feedback upfront and during
You prioritize	Customers help you prioritize
Customers are outside your ship cycle	Customers are inside your ship cycle



Case Study



Enterprise Library – Reusable Code for .NET

- **Reusable code solutions.** Enterprise Library is probably the most commonly used reusable code solution in enterprise .NET projects.
- **Application blocks.** Enterprise Library consists of a number of application blocks addressing cross-cutting concerns such as validation, logging and caching.
- **Standalone or together.** These application blocks can be used individually, or can be integrated together into custom applications.
- **CCE for each release.** The breadth of coverage has made them extremely popular amongst enterprise developers and architects, which in turn has created unique opportunities to use CCE to improve each release.



Enterprise Library – Breadth and Depth

- **Breadth and depth by design.** The customer relationships used in the planning and development of Enterprise Library have spanned from “breadth” to “depth”. This was a very deliberate strategy, which both extremes (as well as the middle ground) providing different kinds of data.
- **Breadth.** Breadth relationships involve gathering data from as many customers as possible. This can provide insight such as “feature X is used in 78% of projects”, although it provides limited insight into how or why that feature was used.
- **Depth.** On the other end of the spectrum, a depth relationship can provide extremely detailed insight into an individual customer’s experience and processes; however there is no way of telling if that user’s experience is typical. It is only through combining data from various points on the spectrum that it is possible to get a complete picture of how customers have been using Enterprise Library.



CCE Techniques Used for Enterprise Library

- Customer Advisory Board
- Prioritization Surveys
- Exit Poll Surveys
- Community Feedback
- Customer Visits
- Customer Advisory Board



Customer Advisory Board for Enterprise Library

- **Customer Advisory Board.** As with other p&p projects, Enterprise Library relies extensively on a Customer Advisory Board.
- **Live Meetings and teleconferences every 2 weeks.** During the planning and development phases, this has typically involved a Live Meeting and teleconference once every two weeks.
- **Context and cross-checks.** A primary value of the Customer Advisory Board is to provide context and cross-checks for data obtained using the other CCE mechanisms. For example, if our exit poll data shows that a particular feature isn't being broadly used, we can ask our advisory board for details about whether they use it, and why or why not.



Prioritization Surveys in Enterprise Library

- **Web-based survey for data.** Before each release of Enterprise Library, we solicit feature requests from customers using as many mechanism as possible (including all of those described in this case study). This can result in quite a long list, and it's very important that the team understands which of these features will provide the most value to the most customers. We get this data by putting together a web-based survey that lists all of the features under consideration, and ask as many users as possible to complete the survey.
- **Simple enough, but rich enough.** It is critical that the survey is designed to be simple enough for users to understand quickly, while also providing rich enough data to be useful for the team. Over many releases, we have tried several approaches, including pure stack-ranking, stack-ranking with “cost caps” and approaches asking users to spend a virtual \$100 across multiple features (allowing, for example, a user to allocate \$50 on one critical feature and spreading the other \$50 on other less important features). While these complex approaches have their benefits, we found that it tended to confuse some users, and the simpler stack-ranking approaches generally provided the highest quality data.
- **Prioritization surveys help establish the feature backlog.** The prioritization surveys are the primary way that we establish the feature backlog for a new release, but other factors (including technical complexity, alignment with other product's roadmaps, and feedback from customer advisory boards) will also influence the prioritization of features.



Exit Poll Surveys for Enterprise Library

- **Exit Polls.** Another form of web surveys we use is “exit polls”. An exit poll survey is targeted at customers who have recently completed a project that uses Enterprise Library, and asks questions on which blocks and features were used, how they were used, and details on what went well and what caused pain.
- **6 months after release.** We typically run an exit poll about 6 months after a major release of Enterprise Library. Exit polls are valuable because they provide broad data on how Enterprise Library is used in the real world.
- **Provides context for other data.** While we do also receive specific feature requests through this mechanism, more often it is valuable in providing context to other CCE data such as prioritization surveys.
- **Inform non-functional requirements.** The exit poll data is also used to inform non-functional requirements, such as how much effort we should put into making the blocks extensible or customizable.



Community Feedback for Enterprise Library

- **Data on real-world usage.** Information from the community is unstructured and difficult to aggregate, but it fulfills a critical role in the middle of the CCE spectrum in exposing data on real-world use of Enterprise Library.
- **Sources of data.** A community is an inherently decentralized beast, but some of the main sources of data are:
 - Forum posts on CodePlex
 - Blog posts and comments
 - Direct e-mails to the team
 - Questions at events such as TechEd
- **Type of data varies over time.** The community is always active, but the type of data mined from it will vary over time. For example, during the planning for a new release the community will be mined for feature requests and bugs in existing releases. During development, the community will be given beta versions and invited to provide their feedback. At release, the community will be monitored for first impressions and to start gathering data for future releases.



Customer Visits During Enterprise Library

- **Customer visits.** While the web provides a great way of getting data from large numbers of people, there is still no substitute for viewing development teams in their natural habitat. p&p team members have visited several teams to gain insight into how Enterprise Library is used on projects, specifically focusing on the different experiences by different team members.
- **Learned a lot.** This approach was used to great effect before Enterprise Library 1.0, where we learned a lot about the experience of using the original stand-alone application blocks.
- **Non-functional requirements.** The data gathered from customer visits lead to a focus on key non-functional requirements such as ease-of-use, extensibility, consistency and integration.



Getting Started

patterns & practices

proven practices for predictable results



3 Steps for Getting Started with CCE

- Step 1. Create a Customer Advisory Board
- Step 2. Identify CCE Activities in your product cycle.
- Step 3. Test CCE, learn and respond.



Appendix

