aMind
ELEVATE YOUR BUSINESS

#JourneyToCTA

My Journey to Become a Salesforce Certified Technical Architect

A Detailed Personal Account with Practical Details for Success

Robert Wing
CTO, aMind Solutions

Retrospective
Handling Q&A
Test taking process
Test taking action plan
Presentation strategies
Preparation recommendations
Insider tips (the good, bad and ugly)

Table of Contents

Welcome to My #JourneyToCTA

This paper describes my personal journey to obtaining my Salesforce Certified Technical Advisor (CTA) qualification. It is intended to pay forward all the help I received from the architect community on my Journey to CTA. In this paper I will share many details that I wished at the time someone else would have posted.

This paper is written particularly for people in companies that do not have a CTA program or significant mentoring from another CTA. This is "what I would have liked someone else to have written about ahead of me" with a bunch of practical details. I don't know that what I've put together is the best answer, but it's what I figured out along the way.

Test Taking Process – CTA Board in San Francisco

Who You Will Meet

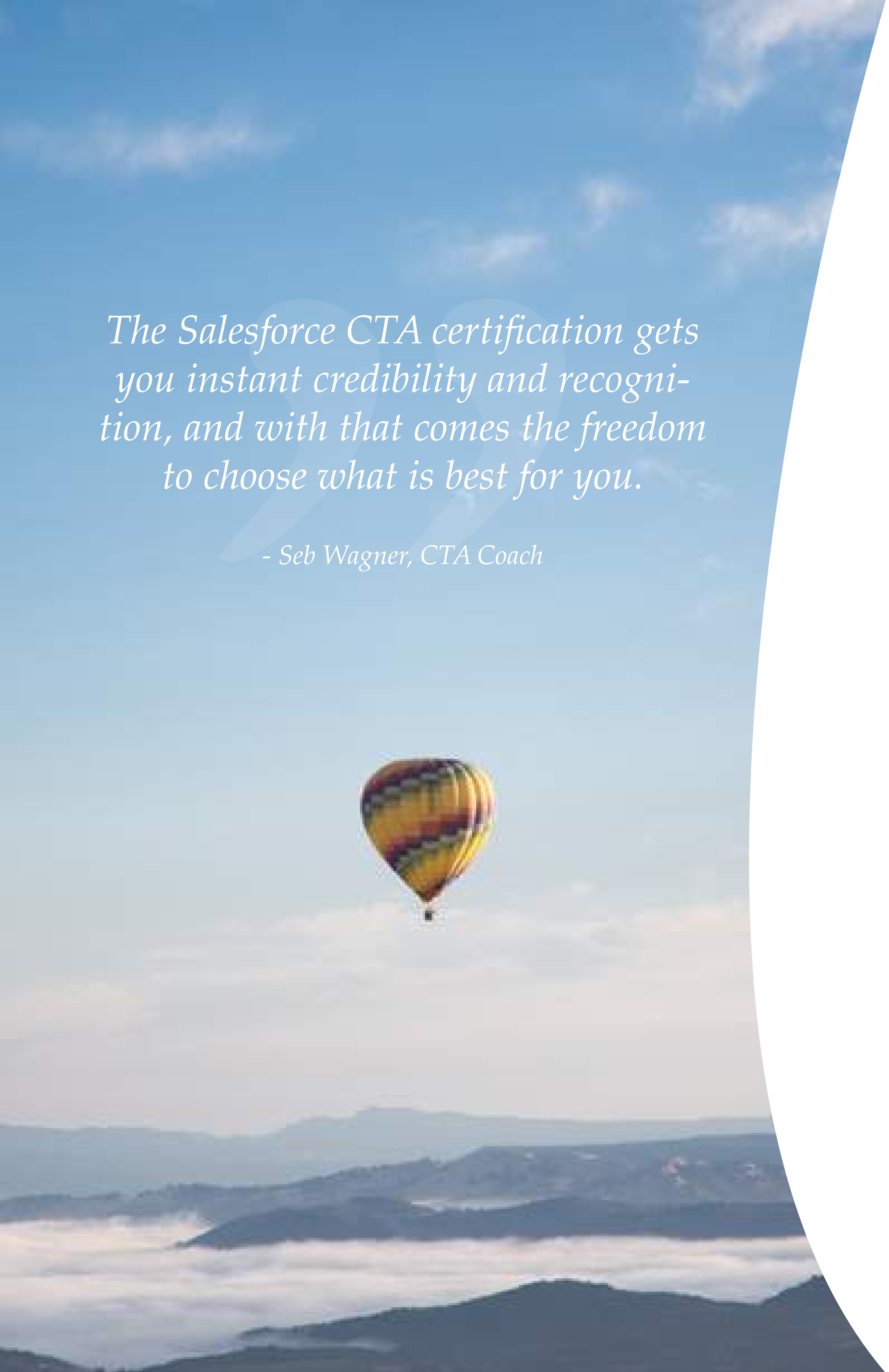
A Facilitator will greet you, take you around, explain everything, facilitate logistics, and probably a bunch of other stuff I didn't even notice. The facilitator will be the main person you will interact with.

While you take the test, the Proctor will sit with you. I had expected them to take away my phone and watch and shut me in a room, but how it works is they sit in the test room with you. Yes, they made me take off my mechanical watch.

Three CTA Judges will evaluate your solution presentation and perform the Q&A.

✓ Checklist for a Smooth Start

- ☐ Arrive in the afternoon or the day before the test.
- ☐ Focus on staying hydrated during and after your flight.
- ☐ Pack a water bottle with you for the test.
- ☐ Pack your favorite pens, highlighters and mechanical pencils.
- ☐ Dress for comfort and in layers. The corporate vibe is on the casual side. The room was hot.
- ☐ Set out everything you will need for the next morning. Check in with family. Stay off social media.
- ☐ Eat a normal, healthy meal. This is not the night to go exotic, or over indulge.
- ☐ Go to bed early and get a good night's sleep.
- ☐ Arise early to set a relaxed pace, eat a big, normal breakfast, and walk to the office or do some exercise to get your body in action.
- ☐ Don't check email or social media. Keep your head clear and focused.
- ☐ Skip or reduce your morning coffee to remain comfortable locked in a small room for 2 hours without the possibility of a break.
- ☐ Arrive 15 minutes early to reduce stress.



Taking the Test

The Facilitator met me in the lobby and brought me upstairs for a quick brief on what would happen before the test started. I had 10 minutes to kill with the opportunity to use the restroom and grab any beverages or snacks from the break room.

We went to the conference room for the scenario prep and met the Proctor who was going to sit in the room with me throughout the test.

When I was ready, and it was open to me to declare when that was (within reason I assume), the Proctor handed me the test with one hand and pressed the start button on the timer when my fingers touched the paper. Timing was very precise.

I worked intensely on the scenario for the 2 hours allotted. The timer buzzed and it was hands-down.

The Facilitator gathered up all the artifacts and took them to the presentation conference room which was separate from the prep conference room; I wasn't allowed to move them.

In my case, I did the prep on a Mac but had to present on a Windows machine because someone else was going to use the Mac for their test. The Facilitator used great care to not accidentally lose the files while moving it all around.

The Test Scenario

Obviously I can't go into specific details of the requirements, but it was HARD; distinctly and materially more difficult in three specific ways than the practice scenarios I had done.

1. It was on the long end of the 7-9-page-plus-title-and-contents expectation set. This doesn't sound like much, but it adds up when you're under the clock.

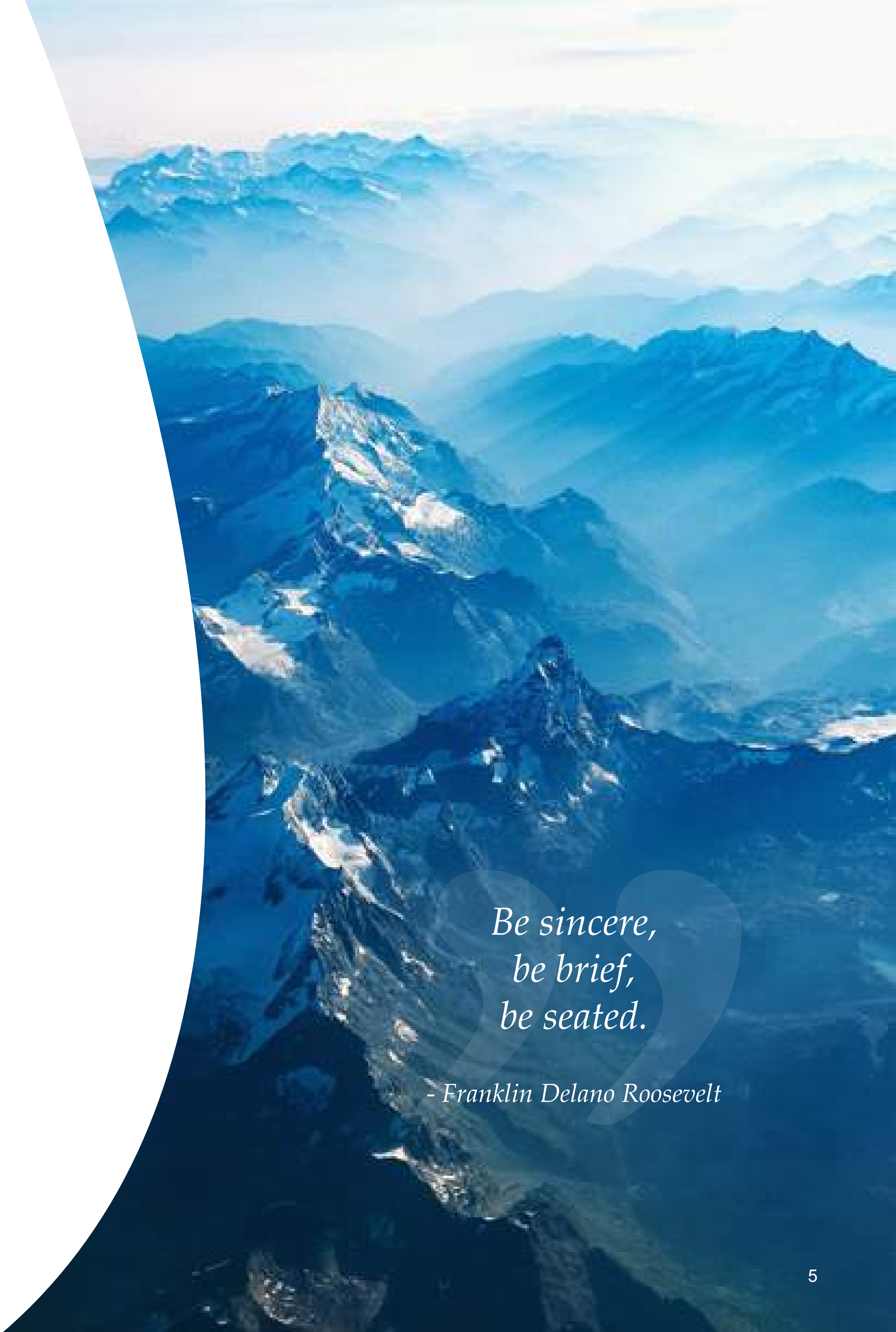
2. It was objectively at least a third more difficult than any of the practice scenarios I'd done in terms of the total number of moving pieces and interested parties and processes and problems to solve.

3. Comprehensive domain expertise was needed to tell the entire story of marketing. From suspect to prospect, sale, onboarding, delivery, support, upsell and cancel lifecycle of the customer. The direct requirements of the scenario were just complicating details of that cycle. The practice scenarios provided more help here to make the presentation easier.

Test Taking Action Plan

The practice scenarios had all followed a reasonably common structure, and I optimized my plan as best I could to match these with a summary, actors, current systems, functional requirements, record visibility + SSO, reporting requirements, process and system and organizational challenges.

My strategy was to create six artifacts to bring into the presentation, one per flip chart paper for the diagrams.



Actors & Apps Diagram

VALUE
Good

Internal and external actors with their Salesforce licenses noted in the left-hand column.

Apps (“Service Console” or “SF1 Mobile App” or …) in the right-hand column, together with their key features / capabilities / solution (“Omni-Channel” or “Live Agent” or …)

Lines connecting Actors to Apps noted with the key processes they were performing with each.

Record Sharing Table in Word

VALUE
Good

- Entity (in scenario terminology)
- Maps to (Salesforce terminology - an org or a spot in the role hierarchy or a standard object or a custom object or a …)
- Record Sharing - the org-wide defaults and any sharing rules or behaviors
- Record Owners - by state as applicable (example: a queue then a person in a role)

Systems & Interfaces

VALUE
Good

Salesforce + Org Strategy + Interface patterns + Packages + Security Strategy.

ESB and ETL and Data Migration Staging

External Applications
- Existing ones that are retained
- Existing ones that are being decommissioned
- New ones added as part of my solution
- Identity providers at the top, keep-systems and new-systems in the middle, decommissioned systems & data archive at the bottom.

External Applications
I’m recommending adding with my scenario, where there were things that didn’t fit well on-platform

Lines between them noting the protocol / pattern / nature of the integrations to implement

Role Hierarchy & Sharing Diagram	VALUE Good
Role Hierarchy - internal and external roles, with attention to implied roles that were needed to meet the sharing requirements.	
Any profile deviations from one-profile-per-actor.	
Sharing Sets - what is shared with whom based on what.	
Programmatic Sharing – what is shared with whom why and triggered how.	
Notes for special cases, for example superusers or admin permissions.	

Data Model Diagram – Main Objects	VALUE Good
Standard objects Custom objects External objects	

Introduction & Summary Presentation	VALUE Waste of Time
• Title slide • Topics slide • Guiding principles slide • Assumptions slide • Risks and mitigations slide	

There were a couple more scenarios that I wanted to create, but I found in practice that I couldn't get them done without sacrificing something more valuable. I relied on my day job experience to be comfortable enough to whiteboard in front of the judges if necessary.

Delivery & Deployment

A couple diagrams to guide the discussion on the dev process & data migrations & test strategy & governance setup & teams & … (Hindsight – experience indicated not creating these during the prep time bit me in the backside hard)

Data Migration & Interfaces Detail

The calls & data moving and how it would be made secure not just that the systems would connect (Hindsight – I'm glad I didn't make these; I wouldn't have had time to use them separately from answering the scenario questions)

Process Flow Charts for the Major Aspects of the Scenario

The calls & data moving and how it would be made secure not just that the systems would connect (Hindsight – I'm glad I didn't make these; I wouldn't have had time to use them separately from answering the scenario questions)

Highlighting for Success

I had a specific highlighting plan for the scenario, to help avoid misses. I brought my own highlighters and this worked very well for me.

I created the presentation before reading the scenario. It didn't have anything scenario-specific in, but I knew what all of these would be based on the practice scenarios at a high level. Topics, guiding principles, and most of the assumptions.

▶ I read the scenario from the start through the functional requirements, highlighting as planned, then drew:
  - Actors and Apps artifact
  - Systems and Interfaces artifact

▶ Next I read the rest of the scenario highlighting as planned, then created:
  - Role hierarchy artifact
  - Data model artifact
  - Entities, Sharing, and Owners artifact
  - Salesforce licenses on the Actors and Artifacts created earlier

▶ I then went back through the scenario front-back and for each item:
  - Noted keyword solution hints next to each item in the margin to help guide me when presenting
  - Cross-checked that solution against my artifacts and corrected any misses or errors

This left about five minutes at the end to check it all over and look for misses and mistakes and empty spaces, and to write fast notes to remember to cover on things I'd be presenting without benefit of backing artifacts.

I was able to create the artifacts I had planned to create within the time limit, and got value from most of them in the presentation.

▶ The two-phase approach to the artifacts worked well:
  - It let me divide the scenario into parts, which made it easier to keep the details in memory
  - Getting the first set of artifacts drawn helped me relax as I started to make progress and focus on the rest.
  - This pattern worked better for me than reading the whole thing front-back which I tried first. When I did that, I had more trouble accurately drawing all the artifacts from memory alone and also found myself only skimming at the end because I was anxious to "start creating something".

▶ The highlighting scheme worked very well:
  - When it was time for creating the artifacts, I just went front-back looking for the appropriate color
  - Doing the highlighting added little time and helped encourage "active reading" where I was hunting for the impactful portions of the scenario
  - I adopted a two-pass approach for each paragraph to emphasize active reading - I'd read it highlight it. I was happy with how this worked out for confirming that I was really getting the scenario into my head. During the presentation didn't have time to fully digest it again.

▶ Two full passes through the scenario I thought added more value than additional artifacts would have:
  - It put more of the scenario into my head, so I was more comfortable with it when presenting
  - It helped iron out a number of small errors and omissions and inconsistencies from the first pass at the solution that would have otherwise tripped up my presentation
  - I tried in practice scenarios getting all of the solution notes with a single pass, in order to save time. The problem was that too often the solution wasn't clear until later information was available which drove too much back-forth while reading and too many misses.

Lessons Learned

#1 Wear more casual clothing made from performance fabrics / dress in layers depending on the season you're taking the test. The room was HOT and dressed as I was that made me sweaty and sticky by the half way point.

I dressed on the corporate end of the corporate casual scale in wool and cotton.

Had I not brought a large bottle of water with me into the room the consequences of dehydration over 2 hours would have made the mistake much worse. -▶ It's easier to concentrate when you're comfortable.

The board was dressed on the casual end of the corporate casual scale so it wouldn't have been weird.

#2 Have a solid plan for managing papers when moving through the scenario and cross-checking. I had hoped to have and had practiced with tables big enough to spread the whole scenario out after tearing the pages apart.

I'd like to think I could have kept eight sheets of paper straight while cross-checking and jumping back-forth across them but that didn't happen.

I read a prior blog from someone who had a big table to work with spreading all the pages out next to each other so I planned for that, but of course corporate conference room reservations can't guarantee that. -▶ I lost valuable time getting the scenario shuffled up and having to dig to find the page I was looking for.

This cost me 60-90 seconds, which doesn't sound like much but when clawing for minutes it would have permitted another simple artifact.

Solution Presentation

The Fastest 45 Minutes, Ever

There was about 20-25 minutes between completing the prep time and starting the presentation. I just sat down and rehearsed the presentation in my head and anything I'd need to voice-over to what I'd been able to draw.

The Facilitator got me and brought me into the conference room for the presentation and Q&A. I introduced myself to the judges. During this period I had an opportunity to arrange the room how I wanted it before the presentation timer started. This turned out to be a training-class-focused room and the initial setup wouldn't have worked well for me with a lectern on one side of the room.

- I moved the flip chart papers around to where I wanted them.
- I tested the PC I would be presenting with to make sure slides and document changed well and I understood how the AV system worked.
- The room had a training-style layout where it emphasized standing at a lectern and presenting slides on the projector. I don't lecture from lecterns and most of my work was going to be with the flip chart papers I prepared earlier plus white board, so I had a minute or two to figure out how I was going to handle the room.

I then presented my solution for the 45 minutes allotted. The judges took notes and checked reference materials as I was presenting. This was a bit different than my usual presentations where the audience is more interactive, but not altogether unlike an RFP presentation where the vendor is permitted undisturbed time to present their solution and the customer take notes.

The fact the judges were actively taking notes registered to me as, "your audience is multi-tasking, figure out what is wrong and fix it". I had to override this instinct in my head and proceed with the presentation.
The Facilitator was also in the room as a proctor through the full test. She stayed quiet and my focus was on the board and I forgot she was there. When a marker died, she was kind enough to grab the bucket with more.

I ran out of time long before I ran out of things to cover. They were serious about the clock - I got to finish the sentence I was in the middle of but that was it.

Discovering My Greatest Weakness

I found the presentation portion to be the most difficult part of the test. I like to think of myself as a professional presenter and I like to tell a good story with well-prepared and relevant supporting materials. One that presents functional solutions in the form of processes for actors, data in the form of life-cyles, and technical requirements in the form of benefits realized and problems avoided against solutions. This simply was not possible.

The problem was that (a) there wasn't time to prepare suitable artifacts to support that, (b) telling a story like that it's tough to make sure you hit every requirement, and (c) I was plowing through a long list of complicated and intricate requirements items, items that I only partially memorized.

How I Could Have Done Better

I was probably too tough on myself for trying to make a very smooth presentation. Watching the provided example of a good test by another CTA he had a number of pauses in his presentation while he organized his papers or thoughts just as I had; that's likely the nature of the beast here due to the time pressure and magnitude of what's being covered.

A better pattern would have been to just plow through it as best I could vs. trying for a smoother delivery, arranging and interpreting questions to make the points I wanted to make in an orderly fashion.

I practiced to just be silent while shifting to the next topic or question rather than 'um' or saying something stupid and having to backtrack; that seemed to work reasonably well. This just followed the old presenting rule-of-thumb that silence may be imperfect but is much better than stupidity.

Lessons Learned

#1 Make better use of the time. You have to present to the scenario, but you do not have to present them in sequence (though presenting them in sequence is easier for you and for the judges if you can make it work).

What I should have done is answer them in a sequence that made sense to me and provided an orderly SINGLE flow through the major content areas against the question.

What I did wrong was first present the topic in an orderly manner, then loop back and tie it to the questions. This would have permitted me to cover more ground in the presentation phase, allowing more solution to be presented and less spill-over into the Q&A.

Cost would have been taking the time to very formally "now we're on X.X.X.X" with each item vs. going in sequence but I think that would have worked better than what I did.

#2 Make sure the the timer is visible from where you are presenting. This would have helped prioritize the last ten minutes or so of the presentation, allowing me to hit more of the specific requirements around delivery and deploy-ment and governance before time ran out.

#3 Ditch the intro slides on assumptions and guiding principles. They didn't add value commensurate with the time spent preparing and presenting them.

Only include the assumptions that change the solution to the scenario. They need to be that important for it to be worth burning cycles on them. These worked best directly with each requirement where the assumption related to, when I ended up re-stating them anyway.

In hindsight presenting the assumptions with each area worked better anyway. For example "here's the best solution for re-porting based on assumptions ___".

#4 Spend less time setting the stage in delivery and deployment, and instead directly answer the questions from the scenario.

I should have gone out of sequence with the questions (or gotten more creative on how they tie to DLC pieces I needed to present), rather than presenting the solution then trying to circle back to the questions.

Solution Q&A

After the presentation portion was a much needed 20-25 minute break. This provided a chance for water, restroom break and a snack.

I used this time to think back through what I had covered during the presentation and if there were points I wanted to particularly focus on correcting during the Q&A.

The Facilitator explained that while I was doing my review, the judges were reviewing their notes to organize questions generally in the interest of my doing as well as possible on the test.

I was brought back into the room and the Q&A started. The questions followed a common cadence:

- Question starting specifically from a point in the scenario
- My answer
- Clarifying points, quite rapid and pointed
- Clear closure of that question and on to the next one from a different judge (this was my insistence on getting them to agree I'd answered the question or not)

When time ran out, I finished the answer to the open question. I could tell the judge wanted to follow up, but was cut short. The better answer he was looking for occurred to me on the elevator ride to the lobby; of course.

I found the Q&A portion to have the feel of a constructive, if intense design review meeting. I didn't get any of the harshness or "toughest hour in Salesforce" or other such things that other people have mentioned in their summaries of the test. I found the judges to be courteous, professional, and with a very highly tuned attention to detail in confirming that I'd FULLY answered all the points in the scenario.

It seemed to be acceptable to acknowledge a mistake and correct - “the solution presented won’t work because ___; the correct solution is instead ___ because ___”. This seemed better than sticking to something wrong. In my case it was two solutions that worked fine on their own but didn’t work together.

When they adjusted the scenario a bit to explore a different area of knowledge (“how would you solve for X.X.X if you couldn’t use ___”) it seemed to be acceptable to quickly describe potential approaches then drive to a preferred option and why that one was preferred, as long as I didn’t bog down in pros-and-cons; they wanted an answer. It’s easier for me to think through it this way, and it seemed OK.

I found them to be helpful when asked (“did I answer your question?”) in explicitly answering either that I’d answered their question fully or that they weren’t satisfied yet and I should go either deeper or pay more attention to a different specific part of the scenario.

The judge’s questions were not at all leading. They did not ask questions like “why didn’t you use ___ for X.X.X” or “are you sure you don’t need another ___ to solve X.X.X”. They just asked for solutions to requirements points or pointed clarifications to them, but there was absolutely no leading to the answer they were looking for.

Time management was again an issue. We ran out of time way before they ran out of questions.

What does it means to be a Salesforce Certified Technical Architect (CTA)?

The Salesforce Technical Architect (CTA) credential is the pinnacle certification for those who demonstrate broad knowledge across multiple development platforms and draw on their skills and experience to assess customer requirements and architecture to design secure, high-performance technical solutions on the Salesforce platform across all areas of domain expertise.

To earn the Salesforce Certified Technical Architect credential, candidates need to successfully pass a Review Board exam. During the exam, candidates are provided with a hypothet-ical scenario with detailed customer requirements and are asked to synthesize the information, design an optimal archi-tecture solution, and present and justify their recommended solution in front of a panel of judges.

Want to learn more? **Check out Architect Journey: Prepare to Become a Certified Technical Architect (CTA).**

Salesforce CTAs are part of an elite group. There are less than 200 worldwide.

- Ashish Agarwal, Salesforce.com Architect and Blogger

Pro Tips

TAKING THE TEST

Check the room setup and rearrange before the timer starts.

- Position flip-chart papers
- Arrange pens, highlighters and mechanical pencils
- Position water bottler
- Clear desk of items you won't be using

PRACTICING

Practice with more and harder scenarios to be ready for a wider range of ways the scenario may be presented to be solved.

Greenhouse was in the same postal code as this scenario, but in my experience wasn't nearly as difficult.

HIGHLIGHTING

Green highlight = actor
Green underline = actor's action
Yellow = entity or object
Blue = an app or channel
Red highlight = a system
Red underline = an interface need

DATA MODEL

I only drew the major objects that had a significant contribution to the solution and the points in the scenario, to avoid creating a cluttered diagram with things like Files, History, Chatter, etc.

PRESENTATION

- Use spare time before presentation to rehearse in your head. Prioritize your 10 minute summary.
- Rearrange the room to fit your style.
- Test the presentation equipment.
- Don't expect interaction. The stage is yours.
- Move presentation timer to where you can see it. It is VERY easy to run out of time.
- "Silence may be imperfect but is much better than stupidity."

Preparation Strategies

This section describes and assesses various steps I took to prepare for the CTA board. I'm skipping details of the knowledge sources (Trailheads,, help, documentation, etc.), as these are well covered elsewhere.

Most Effective Studies

Complete 15 Salesforce Certifications

I completed a total of 15 certifications - the required ones + the optional ones & pre-requisite + Sales Cloud, Service Cloud, and CPQ Specialist

aMind does CPQ so the CPQ was directly relevant to my day-day job vs. for-the-test-specific.

BENEFIT: Very High

I used this knowledge in the test. I didn't use them all, but I used enough of them to be glad I put the work into getting a summary understanding of all of the Salesforce products.

The theory behind this step was that while the required certifications were the most suggested, the test doesn't come with a "that wasn't on the prep list" bail-out, so I needed to be ready against all the products.

3-Page Summary Sheets

I made 3-page summary sheets (functionality, architecture & integrations, implementation work items) for all of the Salesforce products

I didn't complete a certification based on the trailheads and product documentation - Einstein Analytics, Pardot, work.com, etc.

BENEFIT: High

I used this knowledge in the test. I didn't use them all, but I used enough of them to be glad I put the work into getting a summary understanding of all of the Salesforce products.

The theory behind this step was that while the required certifications were the most suggested, the test doesn't come with a "that wasn't on the prep list" bail-out, so I needed to be ready against all the products.

Practice Tests

Practice tests as close to the real thing as possible:

- Timed w/ actual time limits
- Present to a webinar recorder for most scenarios with a time limit
- Present to people live in at least one scenario
- Defend the solution in Q&A to people in at least one scenario

I did:

- Clean Bikes
- Green Roof Systems
- Greenhouse Recycling
- Laptops to Schools
- Reduced size: Galaxy Cars & Pollard Financial

BENEFIT: Very High

Practice tests were critical to finding a balance of what I would and would not be able to accomplish during the CTA exam within the allotted time.

It was also critical to figuring out how to present against the specific requirements in the scenario - it's very easy to get lost and confused trying to present smoothly against something so detailed that you just saw.

I got better at reading a highlighted scenario item and my solution notes in the margin and presenting it as a reasonably coherent sentence. It was tougher than I thought it would be to pull that off in practice and I'm glad I practiced it.

I found presenting into a web meeting recorder to be very good practice - I could spot my own mistakes as I made them and the value was forcing myself to make it a presentation and try to make it smooth. It was distinctly more valuable than not presenting and carried less overhead than arranging a board of colleagues many times.

Practice Tests with a Practice Board

I took Greenhouse Recycling and ran it as a mock board at the office. New to the scenario I gave myself exactly 2 hours to prepare a solution, then shifted rooms and presented the solution to "board members" colleagues, took Q&A for the allotted time, then requested feedback and input on how I'd done.

BENEFIT: Very High

This was the most realistic test of my readiness, and provided the most value in terms of steering remaining prep time. Retrospective: I should have setup the practice "board members" with better guidance on questions, specifically (a) no leading questions and (b) tougher standards for concise and complete answers to questions.

Practice Tests - subsequent runs

I took 2-3 runs through each practice test

1. Whatever happened the first time.
2. Compile lessons learned from that run and do it again.
3. After more lessons from subsequent scenarios, come back and do it again applying those subsequent lessons to practice applying them

BENEFIT: High

This was useful to iron out how I was going to handle the scenario against time pressure, so I could go in knowing what I was going to do and how I was going to do it. It was also helpful for planning how I'd handle time limits.

The value of each practice dropped with each run through the same scenario, because I quickly got to know the scenario and the difficulty dropped considerably each time, but it was still useful to rehearse how I would do things in the test.
Get more scenarios to keep the practice test useful.

The value of the subsequent runs was really working on an efficient and concise delivery of your solutions and major reasoning.

Certification Topic Notes Reviews

I studied the materials covered in the certifications to a level where I could have passed any, or all, of those tests on CTA exam day.

I did this primarily by compiling summaries of my notes from memory, then checking that the summary was accurate.

BENEFIT: Medium

This was good for confidence, but overkill for me. It's the first thing I'd drop in hindsight where time is limited. The level of detail just wasn't possible in the time allotted to the exam.

Vocabulary Lists

I made a long list of 650 terms covered in the certifications representing the knowledge I wanted to have ready on test day. To study I'd go down the list and check that I remembered what each of them was and the items in each category.

BENEFIT: High

I made a long list of 650 terms covered in the certifications representing the knowledge I wanted to have ready on test day. To study I'd go down the list and check that I remembered what each of them was and the items in each category. This was a faster-pass way of reviewing the materials to keep it fresh in my mind. It's useful for me, though perhaps not for others, as one of my memory weaknesses is a tendency to forget the exact names of things I don't use day-day and recall synonyms instead (ex/ 'are they called remote site settings or external site settings?').

Top 5 Preparation Strategies

*****

Complete 15 Salesforce Certifications
Practice Tests
Practice Tests with a Practice Board
Project Sales & RFP Response Experience
Professional Experiences with Difficult
Human-board Examination

Most Effective Experience

Project Sales & RFP Responses

Working with customers early-stage to define the high-level plans, designs, and teams.
This covers:

- The paper responses to RFPs
- "Just state your assumptions and pitch a solution with a cost"
- Presenting an RFP response to a large group in a structured format with both presentation time and Q&A time

BENEFIT: Very High

This seemed to be the most applicable typical experience - starting from a paper set of requirements and pitching a broadly applicable and reasonable solution without a lot of interactive Q&A.

RFP responses were the most relevant experience – both crafting the response and presenting the response.

This would also likely be a good way to gain experience as nobody likes working RFPs so it's unlikely that you'd be refused if you offered to jump on a couple of those grenades.

Project Delivery Responsibility

Direct responsibility for various areas of project delivery - owning specific areas ex/ integration, security, reporting or other.

BENEFIT: Low

These roles are too detailed to benefit the CTA exam. They certainly helped with the pre-requisite qualifications but didn't help much with the broad nature of the CTA exam.

Salesforce Architect Resources

I reviewed most of the library of articles, blog posts, and videos that were posted.

BENEFIT: Medium

Published content is good stuff but knowledge wasn't the problem. While these articles were good for confidence, the exam moved too fast at too high of a level to get much value from how-it-works-under-the-covers type content.

Project Senior Architecture Role

Senior Architects are responsible for the full set of activities and steps needed to get an application out the door, adopted and enhanced.

BENEFIT: High

The experience from broad areas of technical and operational topics definitely helped craft a solution and have a basis for good practices / a "normal" to recommend absent specific requirements.

It hurt in that the role discourages quick-draw-shoot-from-the-hip solutioning which is what the CTA test requires

Navy Nuke (Marine Forged; Navy Honed!)

Military experience with a significant number of board examinations for qualifications with greater consequences to failure.

BENEFIT: High

Experience with other board-style examinations with no-place-to-hide breadth is very useful. This was my closest prior experience and I sampled the same techniques we were taught for that exam heavily.

This is an unlikely experience for many other folks to have – key point is that other professional experiences with difficult human-board examinations should be highly relevant.

Journey to CTA Retrospective

Asked differently: “**this burned a lot of time and money**; **was it worth it?**”

I think **yes**.

- The credential opens a number of doors
- The knowledge and confidence with the full Salesforce platform makes a number of parts of my job a lot easier
- The focus and practice at rapid solution identification and concise & precise explanation of what and why has been professionally beneficial

About aMind Solutions

aMind is a business and technology consulting firm offering unparalleled CPQ and billing implementations. The company specializes in Salesforce CPQ, Quote-to-Cash, Billing and eCom-merce projects. aMind is recognized for:

- 100% focused on quote-to-cash and billing implementations. 100% customer satisfaction. 100's of projects since 2005.
- The right size with the viability of a Big 5 firm combined with the flexibility and expert talent, our "A" minds, of a mid-sized consultancy.
- Trusted by the world's leading brands: Epicor, Red Hat, Manduka, Shentel, Honeywell, Intuit, Thermo Scientific, Valassis, Thomson Reuters, Time Warner Cable and many more.

The company's full-cycle implementation services cover clients from inception to post-implementation support. aMind has a reputation for delivering 100% customer satisfaction, thanks in a large part to our proven agile hybrid implemen-tation methodology.

Service areas include:

- Business assessment
- Solution design and development
- Data cleansing and migrations
- Integrations
- Updates and customizations
- Change management
- Post production support

aMind is headquartered in Alamo, California with office lo-cations in Denver, Colorado and Tbilisi, Georgia.

To learn more, please visit **www.aMindSolutions.com**.

aMind
ELEVATE YOUR BUSINESS

#JourneyToCTA

Robert Wing serves as the Chief Technology Officer at aMind Solutions. He is a manager and architect focused on early-phase software product and project delivery. His focus is on building the right things, building them correctly, and delivering them reliably. As a partner in a ~100-person consulting and software business, Robert does a bit of everything including marketing, negotiation and selling, estimation, planning, contracting, solution design, technical design, project management and governance. Robert's focus is on being "the product guy" and "the details guy" amongst the partners.