DENVER AIRPORT – INTEGRATED AUTOMATED BAGGAGE HANDLING SYSTEM

Iain Simms & Kyle Kennedy
INTRODUCTION

We will discuss:

- Background of the airport.
- Baggage system functionality.
- Why it was needed.
- Project timeline.
- Analysis of the project’s problems.
  - Project management issues.
  - Failures to adhere to good software engineering principles.
  - How these problems could have been avoided.
BACKGROUND OF DENVER AIRPORT

- Covering 53 square miles
- Twice the size of Manhattan.
- 5 full-sized runways.
- 120 planes land an hour with optimum weather conditions.
- 6th busiest airport in US, 10th busiest in the world.
- 22 airlines operate within.
- Denver Airport was used by 35 million travellers in 1997.
SYSTEM FUNCTIONALITY

- Luggage checked-in and placed on conveyer belt.
- Barcode labels are read whilst luggage on conveyer belt.
- The luggage is loaded to a telecar and the barcode scanner informs the telecar what radio frequency to scan for.
- The telecar reaches its destination by following the defined radio frequency.
SYSTEM FUNCTIONALITY 2

- Quickly move all baggage, including transfers, automatically between check-in, the aircraft and pick-up.
  - United Airlines wanted a 35 minute aircraft turnaround.
- Reduce labour costs.
- Be able to fit around the existing airport.
- Be fully automated.
WHY IT WAS NEEDED

- Moving the baggage by the traditional tug and cart system would not have been possible as they are diesel powered.
  - The exhaust fumes would have been trapped in the underground tunnels due to poor ventilation.
  - This would have caused the workers to become ill.
- The scale of the airport was incredible
  - The closest gate was 400 meters from the passenger terminal with the furthest being 1600 meters away.
  - It was essential that moving baggage was done efficiently and as quickly as possible – this could not be performed by a manual based system.
  - Also, the quicker they could move baggage, the less time aeroplanes would spend on the ground – increased airline profits.
PROJECT TIMELINE

- Autumn 1991 - BAE contracted to build baggage system.
- 31st Oct 1993 - Planned opening (missed)
  - Automated system incomplete/error-prone
- March 1994 - System demonstrated to several media groups
  - Absolute chaos, large quantities of luggage and systems destroyed.
- 2nd May 1994 - Faults blamed on lack of testing, not flawed design.
- 28th Feb 1995 - Actual opening
  - Three different systems in operation.
**Problem 1: Ignoring Logplan’s Help**

- Logplan developed the Munich Airport automated baggage system.
  - They had great success.
  - They had the right skills for the job - getting help from Logplan would have been a big step in the right direction.
  - Key to project management: when selecting your team select the right people with the right experience and skills.
- Logplan, as outside consultants, would already have the verified knowledge in place. By avoiding their use:
  - BAE had to pay for time for research to be done (and the research done wasn’t verified as being correct).
  - As there was no previous experience there would have been no one in the team to guide them in the right direction.
  - BAE had no experience in dealing with baggage handling systems before.
    - They treated the luggage as though it was cement, passing it down paths with no concern for its wellbeing.
    - This means that it was not designed by a group of experts and was unlikely to conform to industry standards.
      - Adhering to industrial standards can help ensure accessibility, portability, usability, robustness.
**Problem 1: What they should have done**

- The problems which resulted from the previous points could have been avoided as they were offered the help on a plate.

- Hiring Logplan as outside consultants and having them as key members of the requirements team, planning team and design team in particular. This would have:
  - Given them the clear experience they were clearly lacking.
  - Given them clearer time estimates.
  - Given them much more appreciation for the difficult issues such as the line balancing problem.
Problem 2: Poor Equipment

- Faulty latches caused telecarts to dump luggage on the tracks and become jammed inside the tunnels.
  - BAE solves this issue by modifying every single telecar’s latch.

- Strong airflow often caused the lighter suitcases to fly out of the telecarts.
  - BAE pressure mapped the telecarts in a full-sized wind tunnel to help discover why it was happening and modified the telecarts accordingly.

- Some of tunnels required the telecarts to take very sharp corners (as well as other navigational nightmares) which caused high-stress areas of track.
  - BAE resolved this issue by reinforcing the high-stress areas.

- These issues were caused by poor understanding on the designer’s part.
  - This kind of poor initiation causes delays down the line through rework, errors and omissions.
**Problem 2: What they should have done**

- Resisted starting far too early:
  - “Don't allow the customer to push you into starting the work on the assumption that it will result in an earlier delivery.”
  - This would allow time for correct design and the equipment testing which eventually follows.
  - A full simulation of the system should have been created, testing the full forces which could occur.
    - It would have also acted as a proof of concept, allowing BAE to ensure that the track and telecar design chosen was suitable for baggage transport.

- Carried out proper validation on the equipment chosen to ensure it was appropriate and could perform within the required limits.
  - The validation could have been performed by the Airlines and Logplan, both of whom have experience in luggage transport.

- The faulty latches problem could have been found out through prototyping rather than producing all the telecars, finding out their poor design then having to modify them all.
PROBLEM 3: BAGGAGE SYSTEM = ADD-ON

- Its design had to fit it round existing airport.
  - Design begun after the Airport ‘s construction had begun.
  - The system had to fit around underground tunnels (and any other available space) as it was not planned for.
    - It created ridiculously sharp angles that the telecar cars had to navigate.
    - Had to demolish existing bits of the airport.

- Poor requirements gathering from the Denver officials.
  - It took a demand from United Airlines (7 years into Denver airport project) for them to decide to make all the baggage system automated.
  - Denver Airport assumed that each airline would design their own baggage system.
    - When they didn’t produce the goods, Denver consolidated the baggage systems into one.
    - This was caused by poor communication between DIA and the airlines.
**Problem 3: What They Should Have Done**

- According to expert Bernie Knill:
  - They should have designed the baggage system at the same time as the building.
  - It should have been built at the same time too – preventing construction difficulties.

- DIA’s poor requirements gathering prevented a firm foundation, from which to work, from being established.
  - “Never assume anything” – the foundation of requirements gathering.

- DIA and the airlines should have agreed a communication strategy.
  - By following this communication strategy the confusion over who would design the baggage systems would have been avoided.
Problem 4: Bad Scheduling

- BAE claimed to be able to build the system in 2 years.
  - It really needed 3-4 years minimum.
- This tight schedule introduced a phenomenal amount of human error as:
  - A full simulation of the system was avoided.
  - Full testing of the complete system could not be completed.
  - The training schedule had to be cut right down.
PROBLEM 4: WHAT THEY SHOULD HAVE DONE

- Control the schedule by: defining activities and their resources, sequence and durations then putting these together into a schedule.
- They should never had committed to something they couldn’t deliver.
  - “A considered response when you have had time to evaluate all the factors is much better. A date picked out of the air is good to no-one, least of all yourself.”
  - Work break down structure should have been verified by a consultant as BAE had no previous experience of developing a baggage system – Logplan could have been consulted.
- Full training should have been planned for.
  - Training can identify issues that need to be dealt with
  - Be used to discover areas which need improvement.
- Full testing should have been planned for.
  - Reducing the testing means a reduced testing coverage and less scenarios are catered for.
  - Once you pick a testing strategy – you should stick to it. By not doing so, some requirements may be ignored and not provide a system which the stakeholders require.
- They never accounted for slack time:
  - Tom DeMarco has proved people need approximately 20% slack time during their day to be effective.
Problem 5: Changes

- Changes to requirements/design was carried out by DIA without consulting the airlines or BAE.
- The same situation occurred with the schedule.
- BAE failed to appreciate change impact – this was due to the complexity of the system and its tightly coupled nature.
- As the project was slowly implemented, it swelled in complexity. When design changes were made, this only made things worse.
- Centralized system faults, which ran through many of the baggage system’s subsystems, were discovered.
  - To solve BAE decentralized all of the tracking and sorting computers.
- By far, the biggest change was caused by United Airlines penny pinching. They removed a complete loop of track; having just one loop.
  - Saved $20 million.
  - Required a complicated redesign at a critical phase of the project.
- Other major changes included relocating the outside stations, Continental Airlines larger baggage link and another mezzanine baggage platform.
- Due to time constraints, many of the problems which required substantial redesign were covered over by quick fixes.
  - No time was given to investigation of other solutions.
PROBLEM 5: WHAT THEY SHOULD HAVE DONE

- Established a communications plan:
  - All progress, successes, failures, risk and changes should be communicated to all members of the teams and stakeholders.
  - If something impacts the scope, time, cost, risk, or quality of a task it should be escalated up the appropriate communication channels.

- The system should not have been so tightly coupled and made as modular as possible.
  - If they had followed proper software development conventions and not relied on quick fixes on many occasions, they complexity could have been reduced.

- The additionally requested features could have been solved by better requirements gathering. At no time did the airlines all sit down together BAE and brainstorm everything they’d need.
  - Having all the stakeholders together would ensured none of them forgot features another had requested.

- Quick fixes should not have been used:
  - Full investigation into fixes should have been carried out and then the chosen fix put through the development life cycle.
PROBLEM 6: INSUFFICIENT TESTING

- No full computer simulation – why line balancing wasn’t detected.
  - Line balancing problem discovered 6 months after airport opened.
  - The entire system consists of well over a hundred waiting lines that feed into each other.
- Munich had advised Denver to be very generous with the testing time.
  - Munich had devoted 2 years to testing.
  - BAE ignored this advice.
  - At Frankfurt am Main, getting the automated baggage system tuned to work fairly well took 2 years.
- As BAE had promised a system in far less time than was needed, the testing period was compacted.
  - Furthermore, they underestimated the time required to fix problems – again we see where complexity had run amuck.
Problem 6: What they should have done

- Carried out a full computer simulation of the design, with many different scenarios, before moving on to the implementation.
  - This would have caught the line balancing problem.
  - It could have been used to model the redesign – making it easier to see change impact.
- More time should have been devoted to the testing stage.
  - They were given advice and chose to ignore it.
  - By working together with the Munich airport they could have discovered any parallels in the two systems and anticipated where problems could have lay.
PROBLEM 7: RESERVATION SYSTEMS

- Each of the airline’s reservation systems had their own reservation language.
  - The baggage system had to translate to and from each of those languages.
  - This produced an incredibly complexity.
  - Any changes in the reservation systems had to be catered for.
  - The translation work itself was difficult to do successfully without testing for every different scenario.
PROBLEM 7: WHAT THEY SHOULD HAVE DONE

- These issues come under the heading of integration problems.
  - In order to prevent future problems the airlines should have developed their reservation systems to a common API or at least modified their systems to allow for a common API.
    - This would have made things far easier for the baggage system, eliminating the need for the baggage system to understand all languages.
    - Each of the airlines would only have to understand how to create a common API rather than BAE having to translate many different languages.
    - It ensures that tasks are delegated to the people with the most appropriate skills – the airlines will be in better stead to write APIs where as BAE were not in good stead to write translation modules.
PROBLEM 8: TIMING

- Entire conveyor must wait till leading item can be removed.
- Timing of empty telecars to conveyors therefore is critical.
- Bags need to be distributed to conveyors evenly/sensibly.
- As traffic increases, efficiency can decrease.
PROBLEM 8: WHAT THEY SHOULD HAVE DONE

- Empty telecars need to be distributed to conveyers evenly/sensibly.
- Should have spent much more time on research/design.
- A simulation would have caught this problem.
Problem 9: Poor Staff Training

- Agents issuing on-line baggage tags too quickly caused issues.
  - System incorrectly sending baggage to manual system(s).
  - Assumed the traffic was too busy for the system to handle.
PROBLEM 9: WHAT THEY SHOULD HAVE DONE

- Full training should have been planned for.
  - Training can identify issues that need to be dealt with
  - Be used to discover areas which need improvement.

- 'The more that employers invest in training and developing their employees, the more secure and successful their own businesses will be’.
PROBLEM 10: RELIABILITY OF THE COMPONENTS

- Barcode readers can fail or misread tags
  - (initial 70% failure rate due to printing, 5% afterward).
- Telecars can have radio interference or fail, causing blockages.
- Photocells, even when working, can erroneously report telecar jams.
- Small misalignments can cause telecars to miss or destroy baggage.
- Conveyors can fail, leaving lots of luggage "stranded".
PROBLEM 10: WHAT THEY SHOULD HAVE DONE

- Should have investigated multiple redundancy and considered potential failure scenarios.
PROBLEM 11: UNRELIABLE POWER SUPPLY

- Large power fluctuations caused fuses to trip, stopping motors.
- Electrical engineers never found source of fluctuations.
- BAE had to install industrial power-filters to smooth power-flow.
**PROBLEM 11: WHAT THEY SHOULD HAVE DONE**

- Should have properly consulted the city council to try and resolve this issue.
Problem 12: Politics

- In May 1992 Walter Slinger resigns as head of project.
- Gail Edmond becomes new project head.
- Gail Edmond is not given the same power of autonomy as Walter Slinger.
- BAE access and involvement restricted, resulting in difficulties.
PROBLEM 12: WHAT THEY SHOULD HAVE DONE

- A more experienced management team with real decision making powers should have been assigned from the start.
Problem 13: Disaster Recovery

- No back-up system existed.
- No way to relieve strain from overloaded system.
- No replacement for completely failed system.
- No alternate connecting routes for baggage.
- $50-75 million invested on a "tug and cart" system.
- They should have planned for a disaster recovery
  - “Of companies that had a major loss of business data, 43% never reopen, 51% close within two years, and only 6% will survive long-term.”
Problem 13: What they should have done

- A real alternative system should have been built from the start.
- It should have been defined at the requirements stage.