Cover Oregon was the State of Oregon’s program to implement the federally mandated healthcare exchange. Oregon elected to build and operate its own exchange and decided to go wall-to-wall with the use of Oracle software and used Oracle resources to support the implementation. The website was unable to process any enrollments and the state had to hire or reassign nearly 500 people to process the paper applications Oregon residents were required to mail in. With $300M spent and not a single application taken through the exchange, Oregon and Oracle filed dueling lawsuits in 2014.

### Project Timeline

**2010**
- Obama signs Affordable Care Act (ACA)

**2011**
- Oregon Department of Human Services launches broad initiative to replace healthcare systems.
- Cover Oregon is formed by the State Legislature as a public company (allows for minimal oversight)

**2012**
- Decision is made to save money by **not using a System Integrator**
- Security firm brought in 8 months after being recommended

**2013**
- Security firm brought in 8 months after being recommended
- All Oracle activities are turned over to Cover Oregon, their assessments show poor designs and many problems
- Scheduled go-live is missed.

**2014**
- Lawsuits between Oregon and Oracle begin
- The FBI launches an investigation into possible criminal wrongdoing related to the failed development of the site

### Spend vs. Planned

<table>
<thead>
<tr>
<th>Grant</th>
<th>Projected</th>
<th>Spent</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0M</td>
<td>$150M</td>
<td>$300M</td>
</tr>
</tbody>
</table>

### Key Decisions

1. Taking on a project with a firm deadline – risk on quality or budget
2. Deciding to make the project bigger by combining with replacement project
3. Using a combination of software that has never been used in practice

### Key Takeaways

- Involvement of multiple bureaucratic organizations complicates decision making and results in complex governance
- Audit results should be taken seriously
- Combining projects increases risk

### What went wrong?

Cited in the federal CMS report, this disaster was the result of:

1. Poor Project Management
2. Poor Contract Management
3. Poor Software Management
4. Lack of System Environments
5. Poor Testing
6. No Funding in place to move environment forward.

### Lessons Learned

1. Limit scope or have scope off-ramps when delivery date not negotiable
2. Take action on independent assessments
3. Vendor management & contracting matters
4. Internal talent is a leading indicator of project success

*This is a compilation of publicly available information.*