**Report**

**How Indian Railways can be migrated to AWS??**

**About Indian Railway System**

Indian railways is one of the world's largest railways networks comprising 115,000 km (71,000 mi) of track over a route of 65,436 km (40,660 mi) and 7,172 stations.

All the information technology related activities for Indian railways are handled by one umbrella organization **CRIS (Centre for Railway information services )** set up in 1986. It handles the task of designing, developing and implementing the Indian ticket reservation system. CRIS has developed PRS – PASSENGER RESERVATION SYSTEM providing reservation services to nearly 1.5 to 2.2 million passengers a day on over 2500 trains running throughout the country.

The PRS Application **CONCERT**(Country-wide Network of Computerized Enhanced Reservation and Ticketing) is the world’s largest online reservation application, developed and maintained by CRIS. The system currently operates from 5 Data centers.  
🡪 New Delhi , Mumbai, Kolkata , Chennai and Secunderabad

**Architecture model of IRS**

CONCERT architecture is based on the state of the art technology using 3-tier client-server distributed transaction processing paradigm.

The server clusters are connected together by a core network based on a mesh of 02×02 Mbps leased lines. It is an on-line client server application developed in ‘C’ .  
**Server for hosting** -Blade servers having 64-Bit Itanium 9340 Processor  
**Operating System-** Open VMS 8.4  
**Database**- In-house CRIS proprietary  
**Middleware**-HP Reliable Transaction Router (RTR)  
**Frontend**-DEC Forms, C and FORTRAN as development tools  
**Communication System**s- CISCO routers using 2Mbps DOT lines  
**Network Topology-** Mesh Topology  
**Network protocol**- TCPIP, DECNET

Irctc website whose **frontend**has been developed by**CRIS** and software for running the application is provided by broadvision.

It follows three-tier architecture.

* Presentation Layer—IIS 6.0 on Windows 2003 R2 Enterprise Edition at Web
* Servers Layer (6 servers) with 128 bit SSL GSID Certificate for secure transmission.  
  Application Layer—Broad Vision Enterprise on Windows 2000 Advanced Server ( 40 Servers)
* Database Layer—Oracle 10g Enterprise Edition on Windows 2000 Advanced Server

Server side programming -Asp .NET,Java  
Client Side-Javascript ,Jquery  
Design - HTML,CSS

**Challenges faced by Indian Railway System**

Indian Railways has a huge infrastructure. There are many problems that take place simultaneously. The complex structure and sheer volume of transactions taking place along with other reasons make the IRCTC e-ticketing process very slow. I tried to break down the complex structure of Indian Railways and also tried to find solutions for these problems.

Reasons For IRCTC E-Ticketing Portal To Be Slow:

### **1] High Volumes And Complex Structure**

1] IRCTC is the biggest e-commerce website in the country with more average sales than Amazon and Flipkart combined. And the traffic surges extremely high during [Tatkal tickets](https://railtribe.comhow-do-i-book-my-railway-ticket-online-tatkal-ticket/)booking hours. The traffic is so high that big e-commerce website wouldn’t be able to sustain it.

2] Each year, more than 8 billion people travel through the Indian Railways and the maximum number of tickets are booked through IRCTC website. [Amazon](http://www.amazon.in/)‘s world record was of selling 13.5 million products worldwide, but IRCTC sells over 21 million tickets on any average day. So now you will have a clear idea of the sheer volume of [traffic](http://www.alexa.com/siteinfo/irctc.co.in) that the IRCTC web servers have to handle.

3] The people booking these tickets are only operating on the outermost web layer. There are many layers such as website architecture, server configurations, all the way up to the database. A failure occurring at any of these levels can cause a failed process and the user will have to start fresh from login itself.

4] The whole IRCTC e-ticketing system seems good but the main problem I think is just the enormous demand and supply ratio. According to a statistics, even if the whole system was perfect and error free, only 20 million people would’ve been able to get the tickets and still more than 3 times as many people wouldn’t get a reservation.

### **2] Major Bottlenecks And Preferred Solutions**

### **A] Technical Problems**

#### **Seat selection**

The seat selection process is far too complex than just a straight forward allotment of all the berths in first come first serve manner. There are so many conditions and quotas and seat numbers etc that makes the process even more tedious and complex to handle at the database level.

Railways offer exact seat number/berth to each and every passenger. There can be more that 500 seats on every single train and thousands of trains run through the whole country. So, on a database level, after applying quota’s and other conditions and also preference by the passengers, it becomes very difficult to manage.

The solution to this would be to just sell the tickets and let the passengers choose the seats/berths later. This system is used in commercial airlines and has proved to be very efficient. The only problem with this would be to educate the people who travel through the train, which will take some time. Some improvements can be done as to retain the berth preference in this case. But it will surely reduce the stress and greatly improve the performance of the system.

#### **Payments**

This is another huge bottleneck. Payment process triggers a series of simultaneous process and complex handshakes between the IRCTC and payment gateway systems and banks and security measures etc. Failure in these processes may literally cause the user some actual money. But IRCTC does [refund](https://railtribe.comwhat-are-the-irctc-refund-rules/) money in such cases of failed transactions where money is deducted by no ticket is allocated.

Indian Railways is already working on a “mobile wallet” like the concept, where people will be able to keep cash in a prepaid way. This way, the IRCTC e-ticketing process will get a whole lot faster as well as think about the extra money that will stay with the railways with no interest to pay. This will help the Railways economically also.

Maybe they can provide credit cards like facility also. Many companies are doing this currently.

#### **Database**

All of this systems boils down to the database architecture and data handling techniques. They could beef up their architecture with more replication and sharding capability. Most of the work is read bound, so this can be improved in that process.

If the application and database layers are properly coupled then it may be possible to implement active caching which will minimize the rapid read impact without losing the validity of the data being processed.

Most of the irregular traffic load comes during the booking of [Tatkal Tickets](https://railtribe.combook-tatkal-ticket-irctc/). This is a small fraction of the huge data that is open for booking. A special type of highly optimized database should be developed to handle this traffic. Also, existing apps should be fine tuned to execute this process more efficiently. Additional hardware should be made available to handle this sudden surge in traffic.

#### **Application Design and Architecture**

It’s high time that the Indian Railways get rid of the old age software systems and install fresh new technology. IRCTC mostly uses Windows Stack. I think they should shift to Unix based environments as Unix is stable and runs smoothly without any failure’s.

Also, deploy Cloud services to handle that extra load at some times. The interface is too crowded, therefore, remove unnecessary tabs or put them together in a single tab. A clean interface is very important for smooth functioning. Improve the API by breaking it down into different modules and setting up teams that will look after each individual module. Therefore, maintenance will be easy and efficient.

### **Non-Technical Challenges**

At the end of the day, Indian Railways is a state carrier. The flaws of our bureaucracy do affect it too.

Slow decision making, scarcity of experts in hardware and software teams. This affects the overall improvement and maintenance of IRCTC. Lack of will is also a major issue. Also, corruption is also a huge roadblock that is keeping IRCTC to grow at a faster rate.

Lack of funds and the Top Down culture that we have here also had a bad impact on the growth. People sitting on the top, if uneducated, don’t know what is lacking or what needs to be improved, therefore, do not initiate the process to improve various things.

**IRS System on AWS**

**Services used to build IRS model on AWS**

1. **AWS Simple Email Service:-**

* Amazon Simple Email Service enables you to send and receive email using a reliable and scalable email platform.
* For Email, We can use the Simple Email Service of AWS.
* This Service is highly reliable, secure and available we can rely on Email Service for our email Conversation.

1. **Simple Notification Service**

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.  
   
for SMS, we are going to use Simple Notification Service.

1. **Amazon EC2**

* it is a highly available secure virtual machine service of amazon which can host a lot of operating Systems including redhat, ubuntu, windows and their different versions of these servers.
* We are Going to use EC2 for hosting our web server and App Server, and management Servers with Auto scaling and Load balancers.
* Autoscaling groups is used to scale virtual machine by their size or in number of instances, it can create and destroy a machine automatically as needed and it a machine gets terminated by an error then it create another instance with same configuration
* Load Balancer: Load Balancers Are used to distribute the traffic to the ec2 instances.

1. **Amazon CloudFront**

* Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications, and APIs to customers globally with low latency and high transfer speeds.
* We are going to use AWS CloudFront to improve User experience by using Amazon Edge Locations to cache the content.

1. **AWS Lambda**

* AWS Lambda is a serverless compute service that lets you run code without provisioning or managing servers, creating workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes.
* we can use lambda function to store employee related details and attendance details to save in Amazon dynamoDB with the help of SQS.
* For  Error Reporting Like Train running late, change of platform we can use lambda functions to acknowledge the other train authorities.

1. **Amazon Relational Database Service**

* Easily set up, operate, and scale a relational database in the cloud.
* We are going to use RDS with MySQL  Database with Elastic Cache enabled too**.**

1. **Amazon DynamoDB**

* DynamoDB is a fully managed, key-value, and document database that delivers single-digit-millisecond performance at any scale.
* Error Reporting DB we going to use  AWS DynamoDB. with Lambda trigger. With Amazon SQS Service**.**

**Summary**

So by using all the above mentioned services , we will create architect model and make a fully functional IRS website and can migrate the existing website.