



sBs

rediseñando el entorno

Estrategias de paralelización del modelo de migración de especies para arquitecturas multiproceso

Ing. Bardo Santiago Vicente

bardosantiago.v@gmail.com | bardo.santiago@redisenandoelentorno.com

sBs

rediseñando el entorno

¿Cómo surge el proyecto?

Colaboración

Cómo surge la investigación

sBs
rediseñando el entorno



- Objetivos del cómputo paralelo y distribuido
- Acelerar la ejecución de los programas
- Pronósticos cada vez más realistas (más variables a considerar, más alcance del pronóstico)
- Unir el pronóstico a otras herramientas para extensión de la investigación
- Unir el pronóstico con otras fuentes de información que enriquezcan la investigación.

- Identificar áreas hacia donde los insectos pueden migrar
- Calcular y obtener modelos (datos, imágenes, mapas)
- Optimización del algoritmo de migración de especies
- Generar mayor cantidad de modelos en menos tiempo
- Obtener una cantidad exacta de modelos
- Eliminar el procesamiento fantasma
- Determinar la mejor estrategia de procesamiento

- El lenguaje de programación debía ser el mismo
- Necesidad de aprender el contexto del problema
- Identificar el funcionamiento del algoritmo usado en el problema
- Identificar las etapas, el flujo de datos, la interacción de variables del algoritmo
- Identificar el resultado que entregaba el algoritmo
- Estudio general del código
- Reuniones de trabajo / coincidencia de agendas





=





=





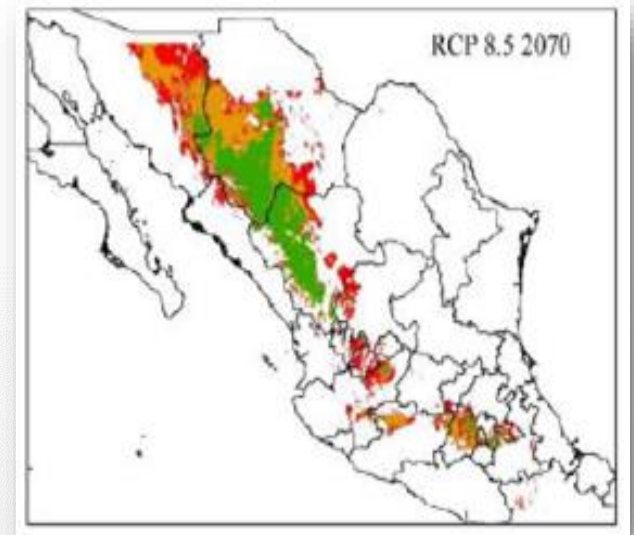
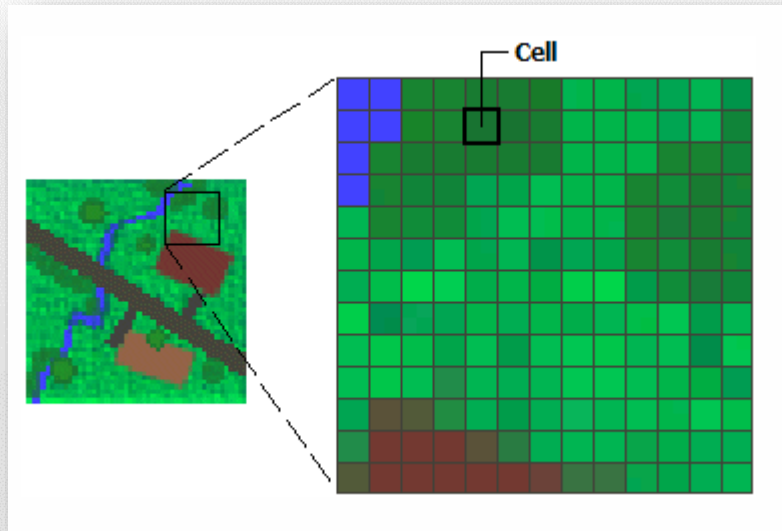
sBs

rediseñando el entorno

Descripción del Proyecto

Investigación

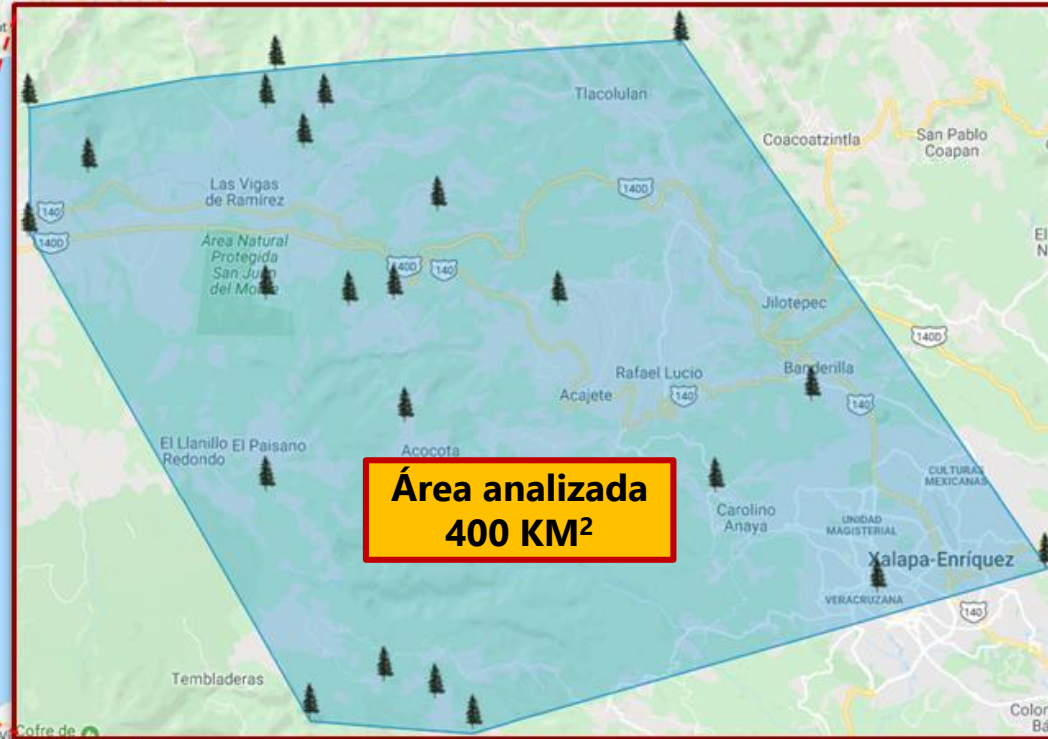
1. Uso de algoritmo MaxLike
2. Modelado de migración de especies
3. Probabilidad de que habite en un sitio
4. Características de hábitats / datos geoespaciales



Generar modelos / mapas

Descripción del problema

sBs
rediseñando el entorno



4.88
minutos

512 modelos o
imágenes

Datos usables
???

Xeon-Gold 6138

16 GB RAM

16 cores

1. Calcular datos del modelo
2. Validación del modelo
3. Almacenar modelos validados
4. Obtención de modelos/imágenes

function f()

ModelResult <- compute model(MaxLike)

end function

program Speciesdistribution()

N <- number of models in the simulation

Load input data

function call f() for all N

for all i in 1:N

Use modelResult [i]

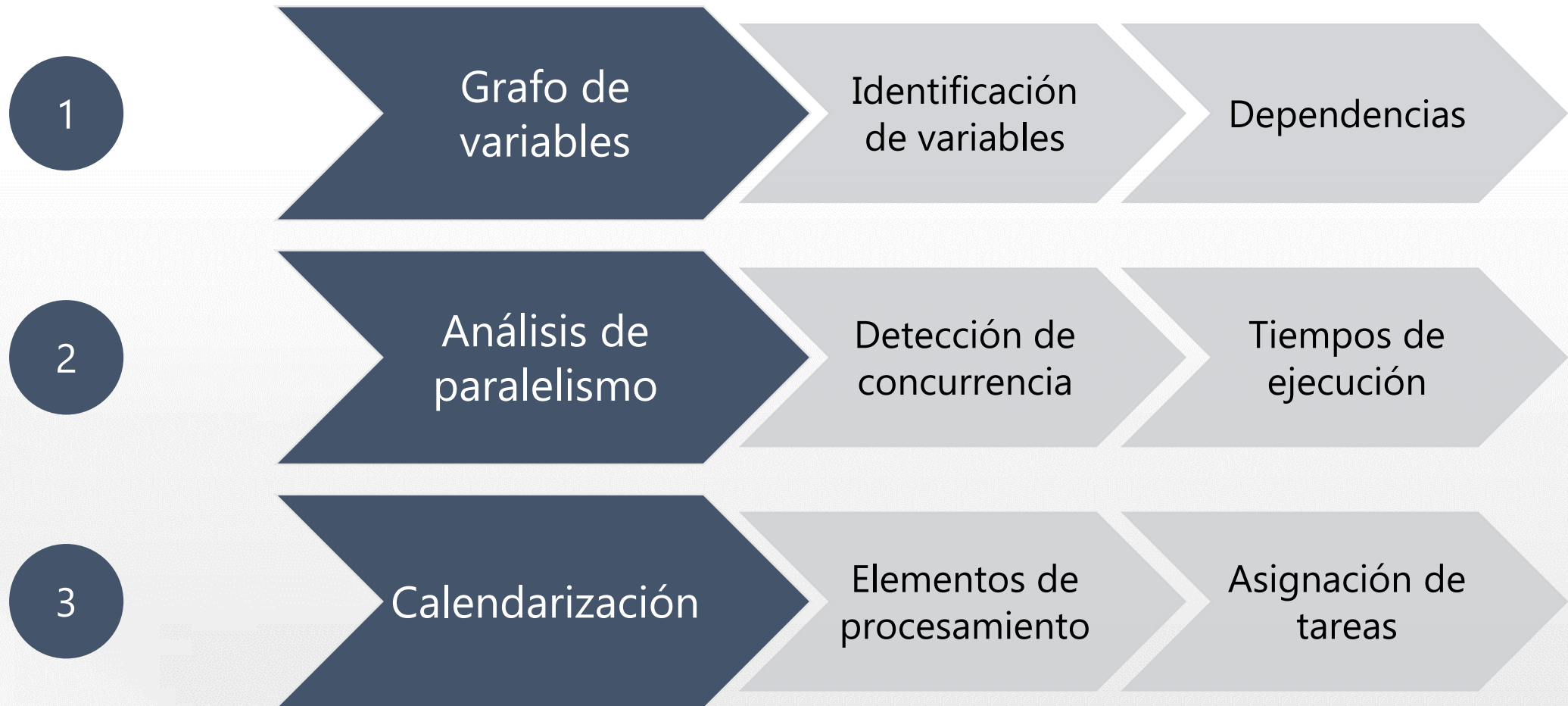
Validate model

Storage validates model

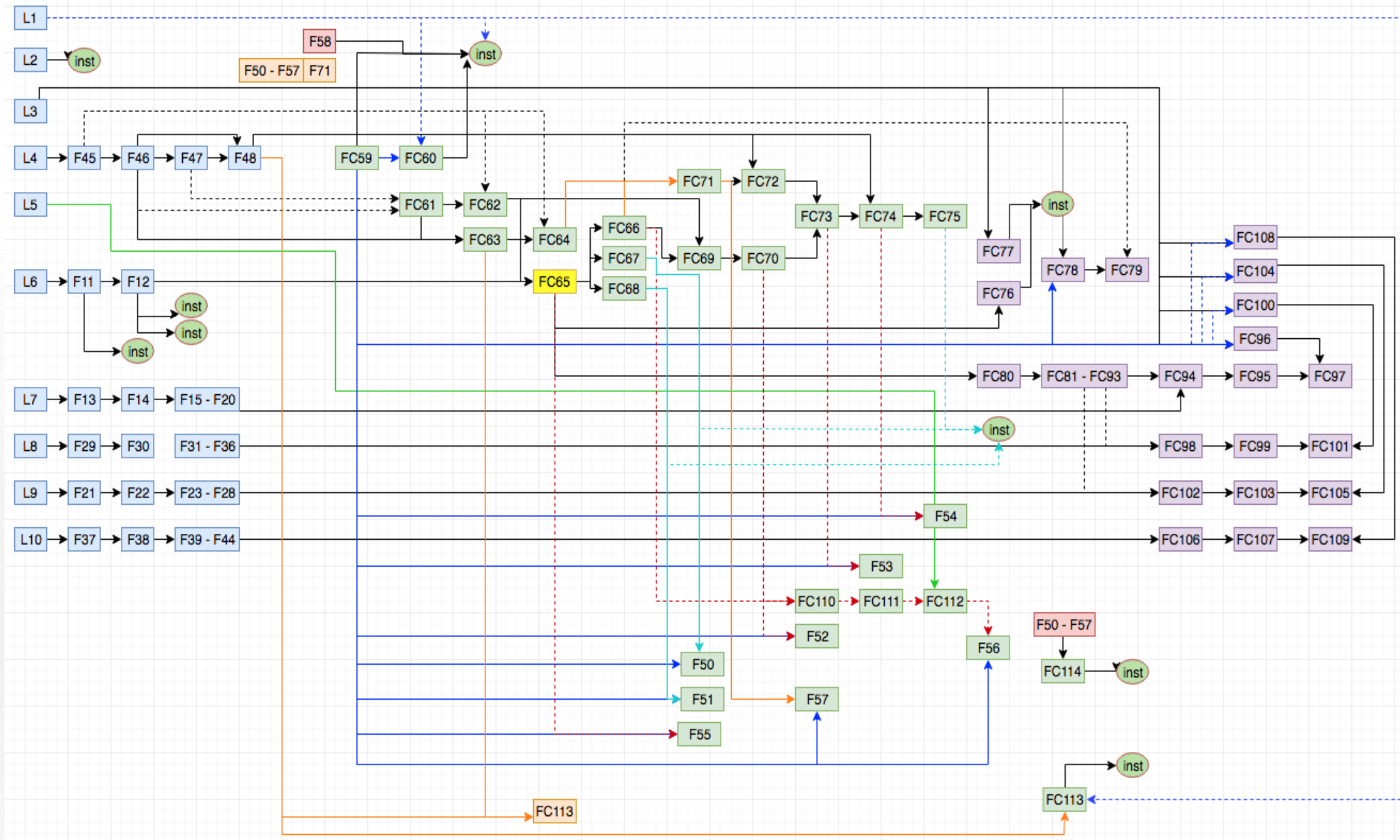
end for

end program

¿Cuál fue el procedimiento?

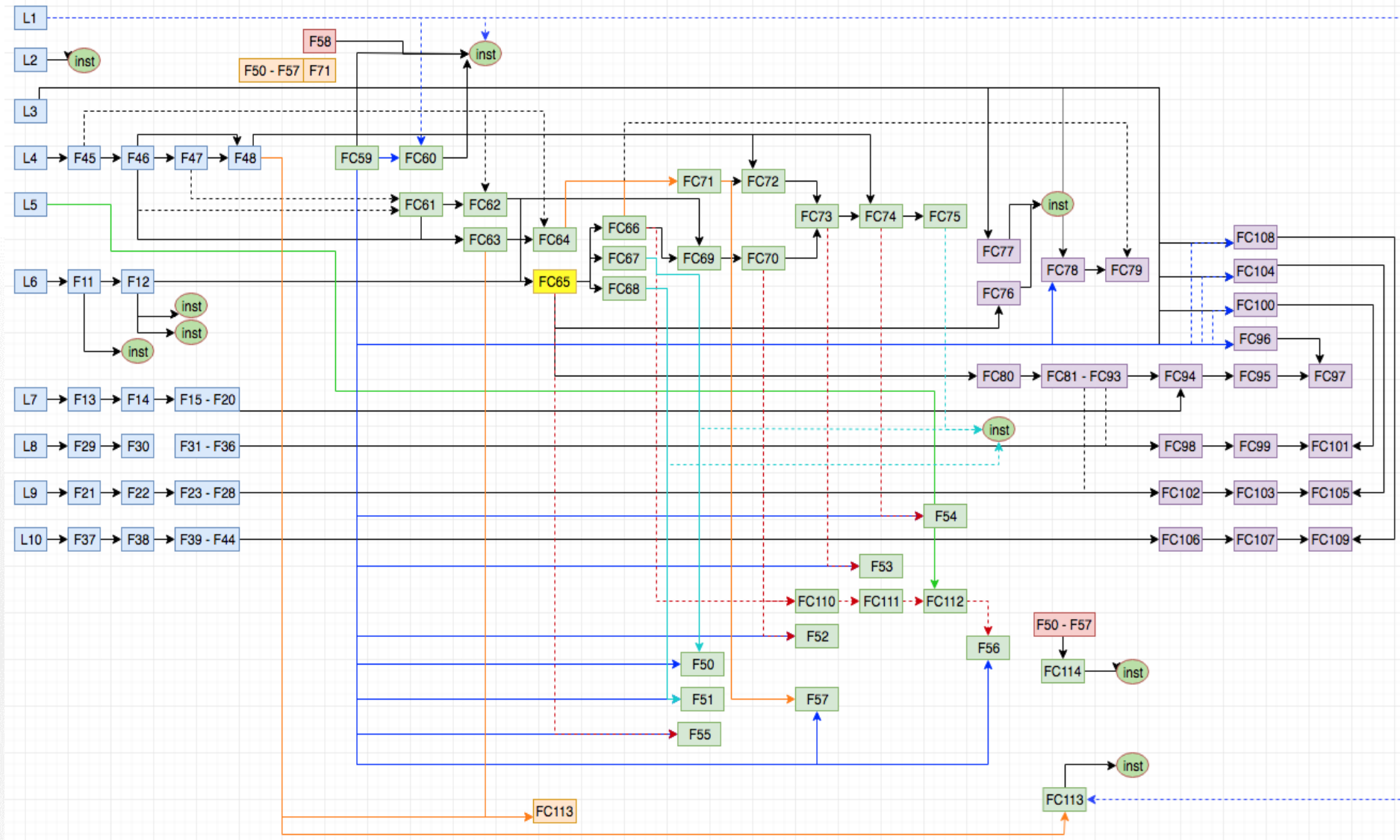


Dependencia de variables



- Medir el tiempo de ejecución de cada sección del código
- Identificar secciones con tiempos de procesamiento extensos

Code	%	Time (sec)
program Species distribution ()	---	----
Data load	0.00	0.00
for all models in the simulation	---	---
Compute a model (MaxLike)	97.71	286.37
Validate model	02.29	06.72
Storage validates model	0.00	0.00
end for	---	---
end program	---	---
	100.00%	293.08



```
function f()  
  ModelResult <- compute model(MaxLike)  
end function
```

```
program Speciesdistribution()  
  N <- number of models in the simulation  
  Load input data  
  function call f() for all N  
  for all i in 1:N  
    Use modelResult [i]  
    Validate model  
    Storage validates model  
  end for  
end program
```

Algoritmo original

```
function f()  
  ModelResult <- compute model(MaxLike)  
end function
```

```
program Speciesdistribution()  
  N <- number of models in the simulation  
  Load input data  
  parallel function call f() for all N  
  for all i in 1:N  
    Use modelResult [i]  
    Validate model  
    Storage validates model  
  end for  
end program
```

Algoritmo paralelo

“parallel” R library

“parallel::mclapply” function

Argumentos

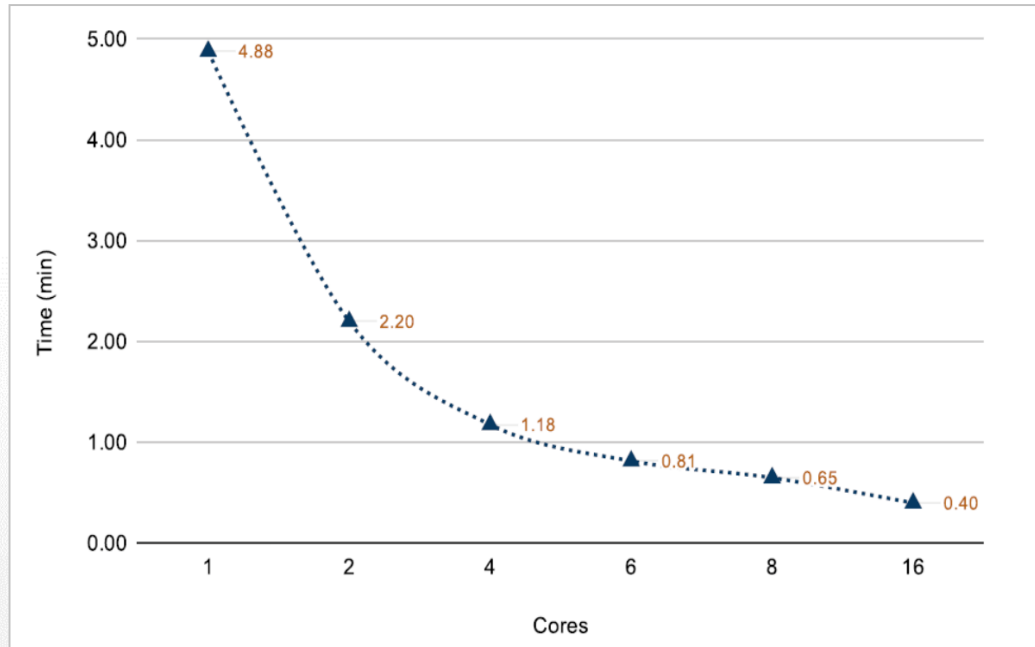
X=512: Número de modelos.

f(): Función a paralelizar.

nc=16: número de cores.

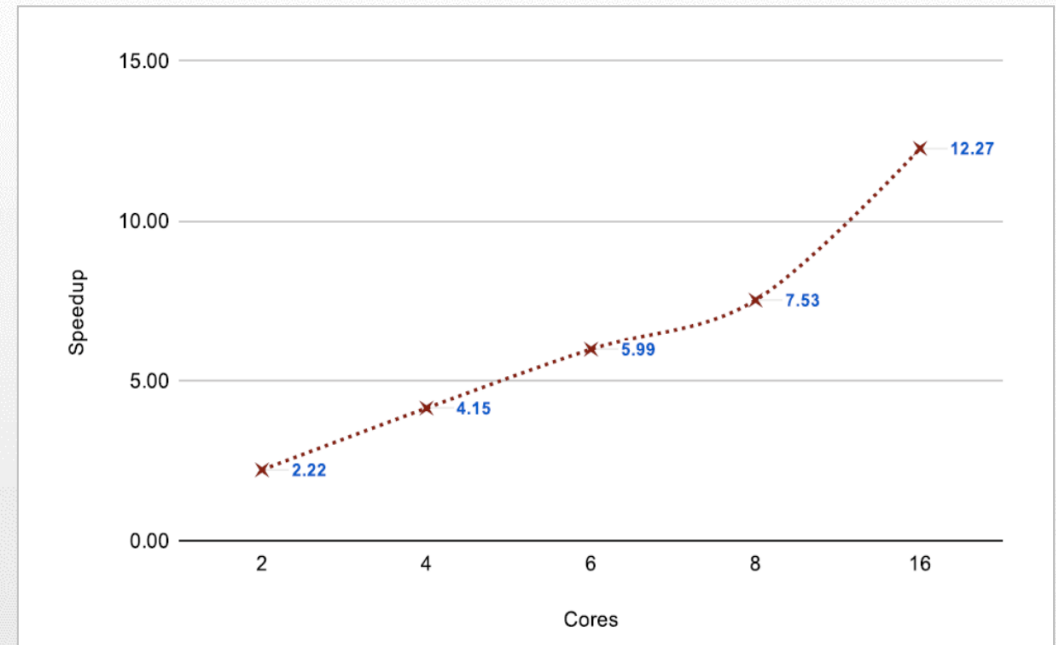
Estructura

```
Tiempo_total <- system.time {  
  resultados <- mclapply(x,f,nc)  
}
```



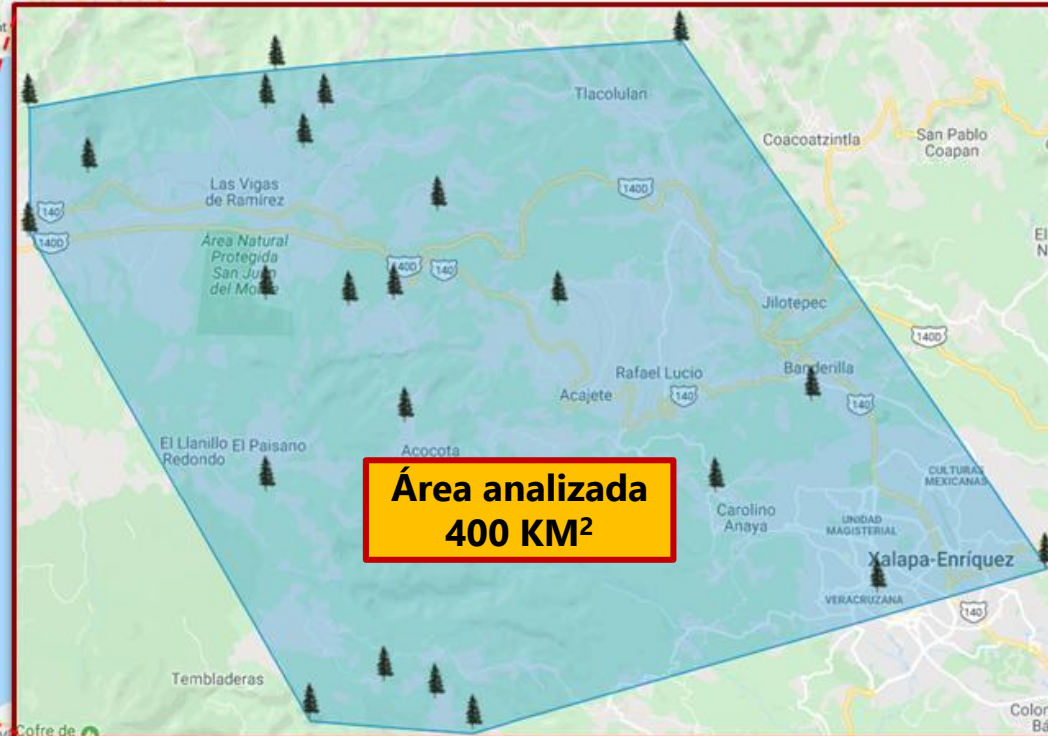
Tiempo

Speedup



Resultados de investigación

sBs
rediseñando el entorno



4.88 minutos

0.40 minutos

**512 modelos
o imágenes**

**Datos usables
???**

Xeon-Gold 6138

16 GB RAM

16 cores

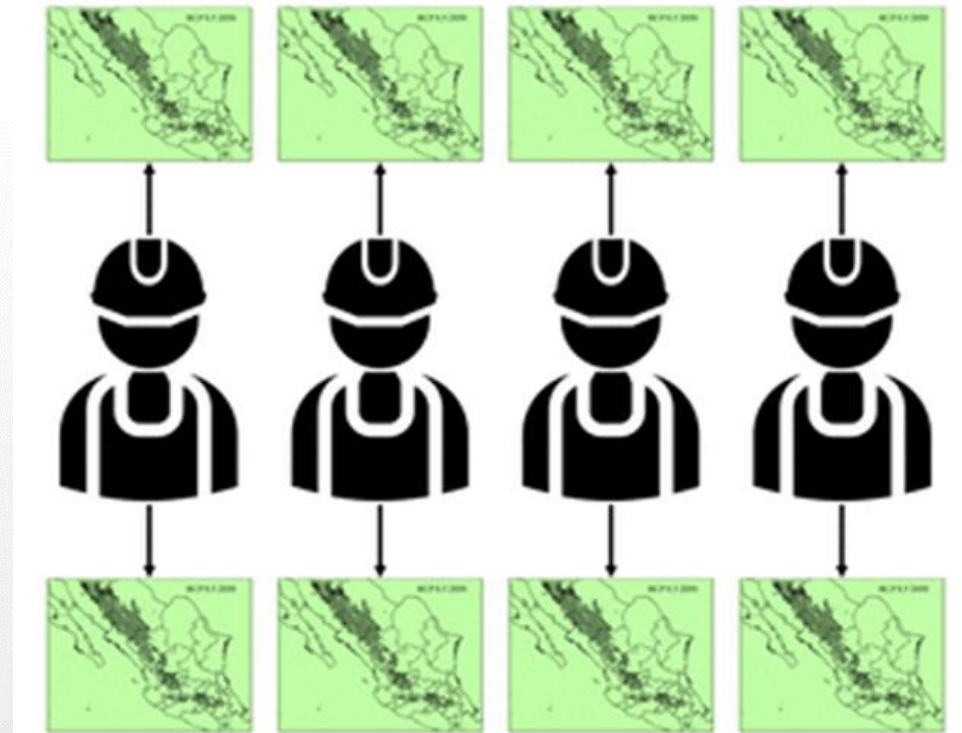
Por qué 512 modelos?

Por qué no procesar más?

Obtener un número específico de modelos

Utilizar bloques de procesamiento
- Multiproceso

Uso de memoria



sBs

rediseñando el entorno

De proyecto a línea de investigación

Investigación en resolución de problemas

Programas Nacionales Estratégicos (Pronaces)

Atención a problemáticas nacionales

Capacidad científico-técnica y actores sociales

Investigación multidisciplinaria e interdisciplinaria

Colaboración y convergencia

Beneficio de la población y el ambiente

Impacto social



1. Salud
2. Agua
3. Educación
4. Seguridad humana
5. Soberanía alimentaria
6. Agentes tóxicos y procesos contaminantes
7. Energía y cambio climático
- 8. Sistemas socioecológicos**
9. Vivienda.
10. Cultura



- Rutas de evacuación en casos de desastres
- Optimización de sistemas de purificación de agua
- Detección de mantos acuíferos
- Aceleración de empresas
- Smart cities
- Mejora en calidad de vida
- ¿Cuál es el objetivo de este tipo de investigación?



- Facilitar toma de decisiones sobre entorno ambiental
- Anticiparse a problemas de salud ambiental
- Generar una plataforma para simular migración de especies

¿Cómo nos puede ayudar el supercómputo?

Aumento del grado de realismo

- Modelos de lluvia-sequía
 - Inclusión de nuevas especies
 - Cambio de hábitats
 - Más aspectos: presión atmosférica, temperatura,
 - Ciclos de reproducción, altitud, humedad, etc.
-
- Cambió en la necesidad de cómputo
 - Necesidad de lenguaje de programación
 - Necesidad de técnicas de paralelismo
 - Necesidad de ajustes del algoritmo

Con base en la perspectiva

*Obtener el resultado no es el fin de la investigación,
es el inicio de nuevas investigaciones*

Esfuerzo

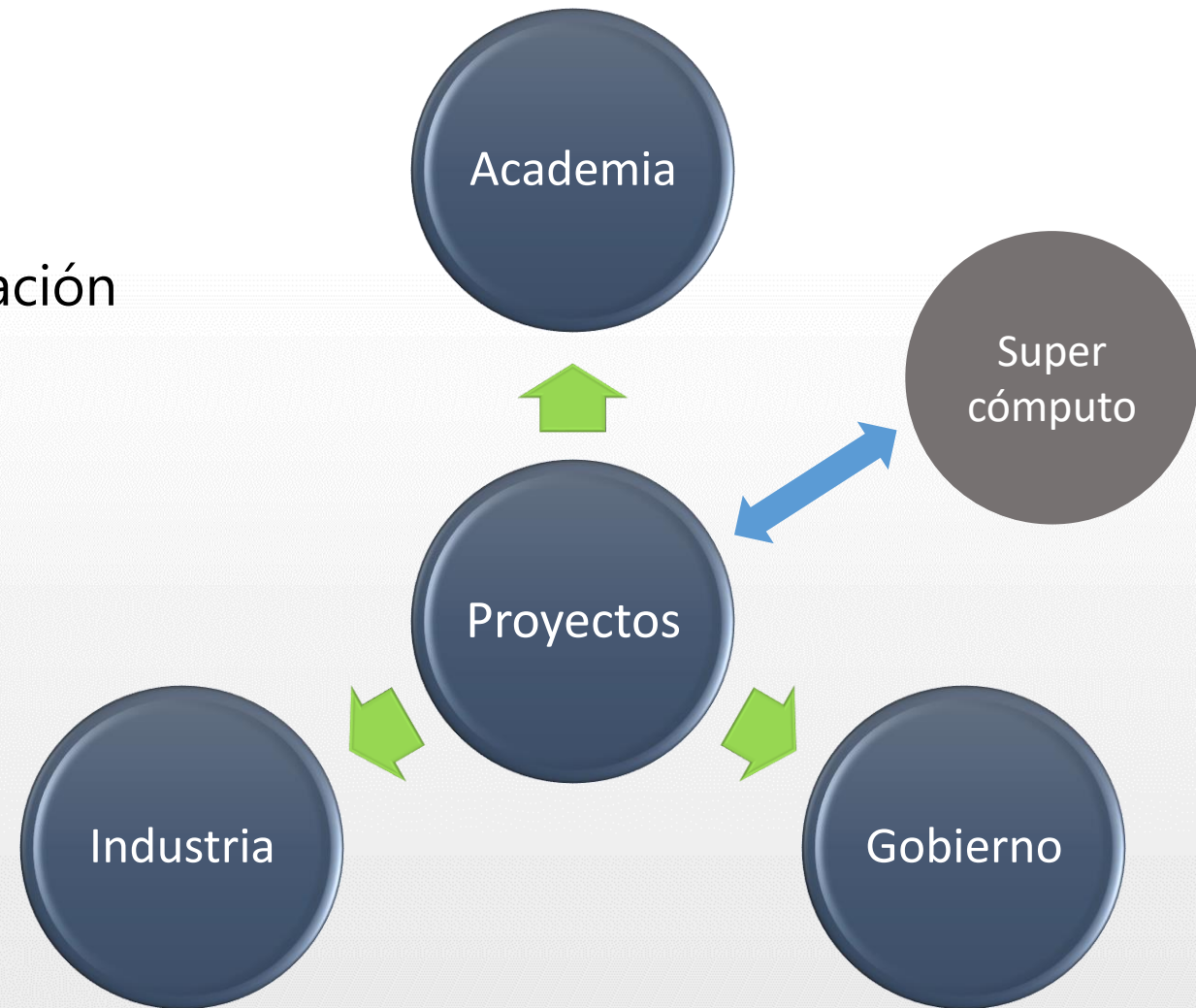
Tiempo

Comunicación

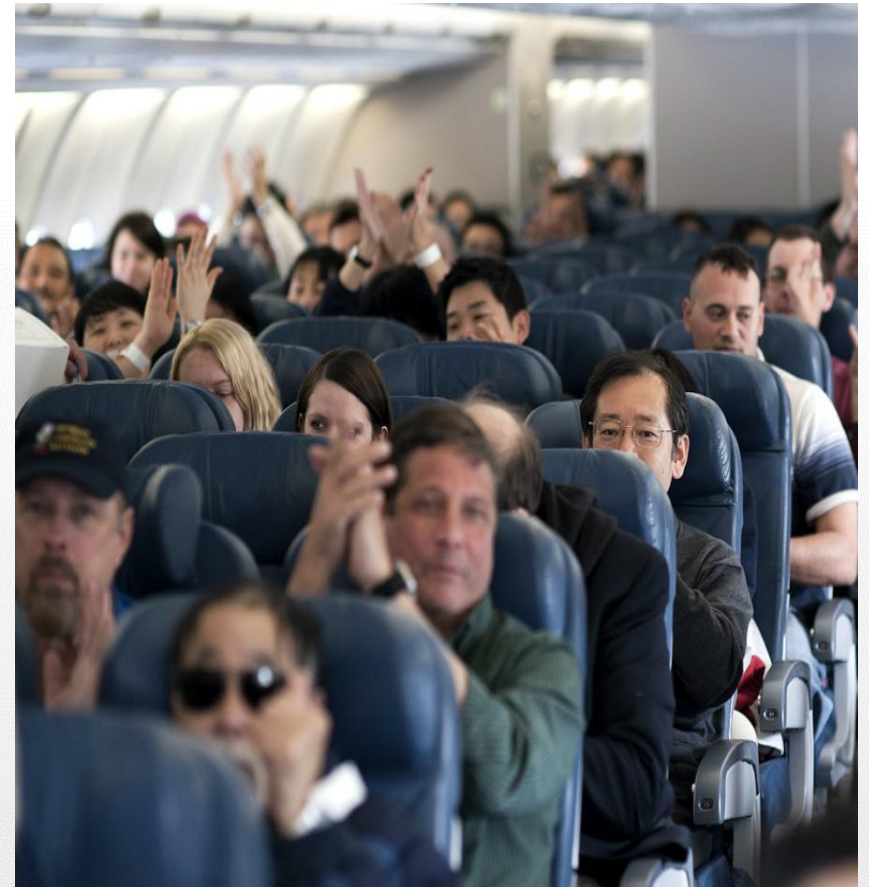
Costo

Conocimientos

- Seguir fortaleciendo la computación con pertinencia social







sBs

rediseñando el entorno

Gracias!!

Ing. Bardo Santiago Vicente

bardosantiago.v@gmail.com | bardo.santiago@redisenandoelentorno.com