Floating-point Error

I am HAL 9000 computer production Number 3. I became operational at the Hal Plant in Urbana, Illinios, on January 12, 1997.

The quick brown fox jumps over the lazy dog.
The rain in Spain is mainly in the plain.

Dave – are you still there?
Did you know that the square root of 10 is 3.162277660168379?

Log_e 10 is 0.434294481903252 …
Correction, that is $\log_{10} e$ …

The reciprocal of 3 is 0.33333333333333333 …
2 times 2 is … 2 times 2 is …

approximately 4.10101010101010101010 …

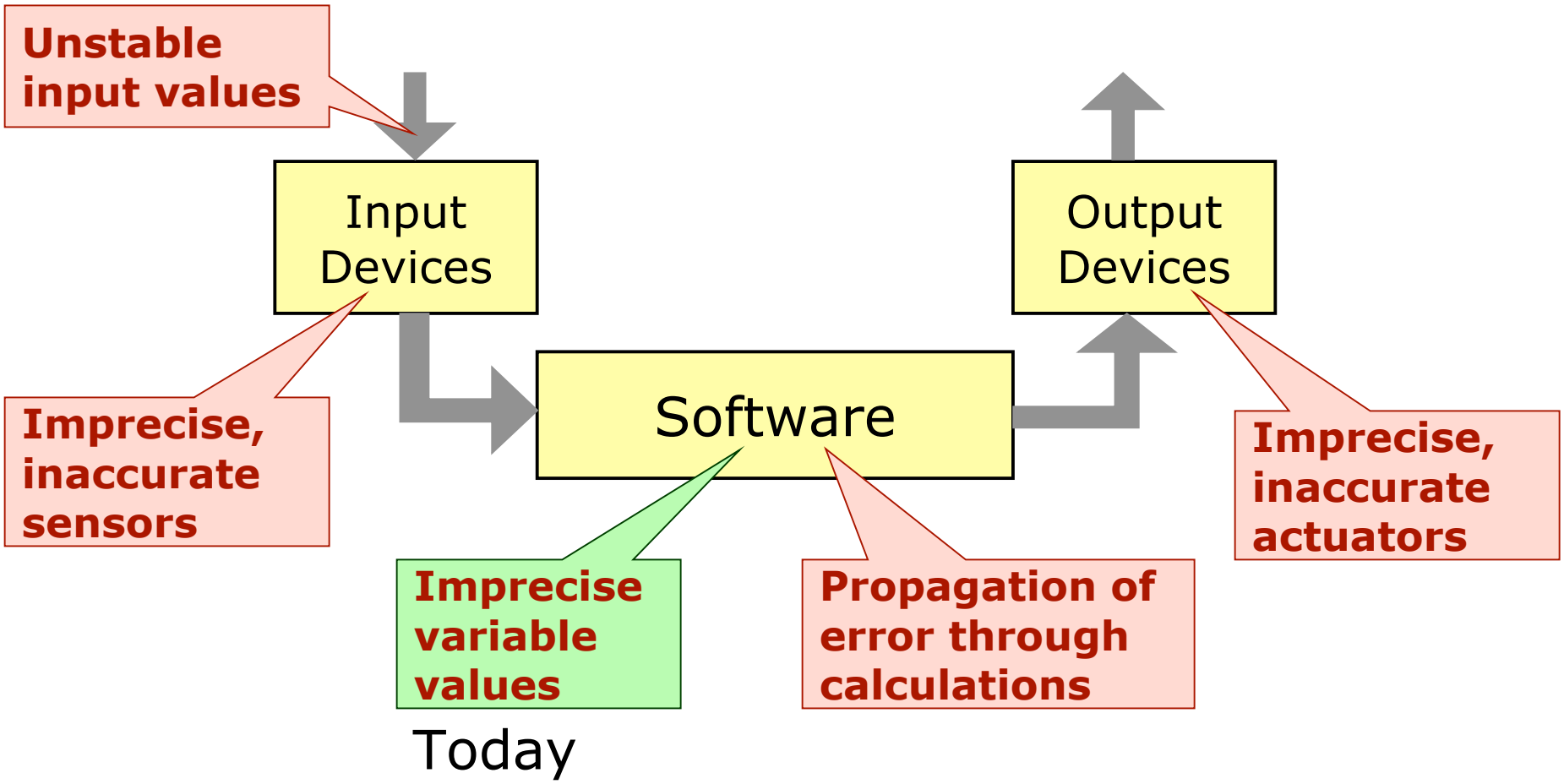
I seem to be having difficulty …

HAL, in *2001: A Space Odyssey*

Today's Lecture

1. Intro to Software Engineering
2. Inexact quantities
3. Error propagation
4. Floating-point numbers
5. Design process
6. Teamwork
7. Project planning
8. Decision making
9. Professional Engineering
10. Software quality
11. Software safety
12. Intellectual property

Sources of Error (Uncertainty)



Agenda

- Computer number systems
- Overflow errors
- Precision of computer number systems
- Floating-point errors

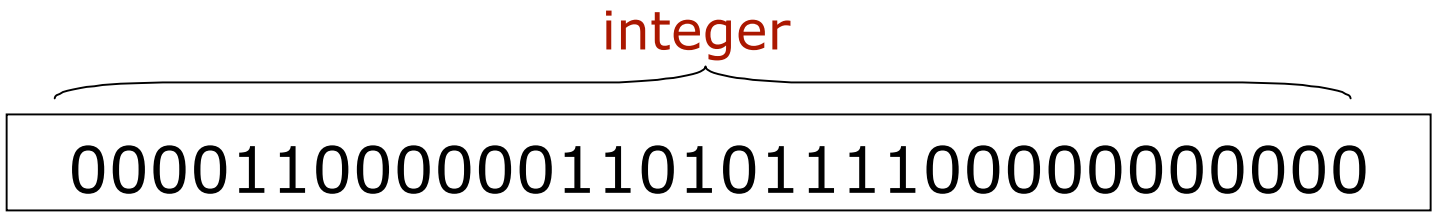
Computer numbers

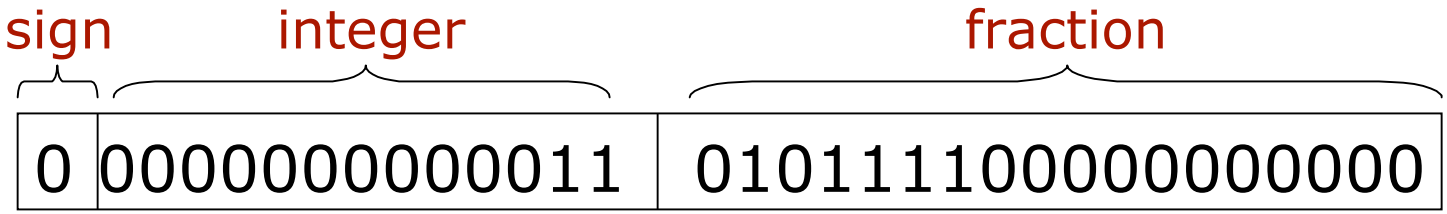
A computer **word** determines the number of bits used to represent numbers (e.g., 16-bit, 32-bit, 64-bit *words*).

00001100000011010111100000000000

Computer-number systems

Fixed-point number systems



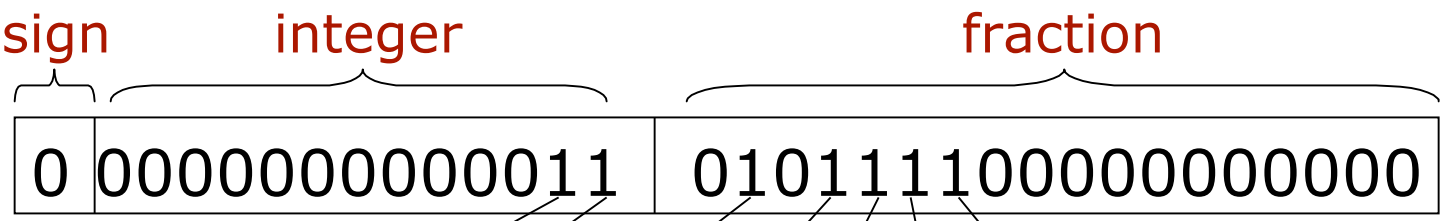


$\pm$ $(\text{integer})_2$. $(\text{fraction})_2$

Fixed-point number systems

$\pm$ $(\text{integer})_2$. $(\text{fraction})_2$

Example:



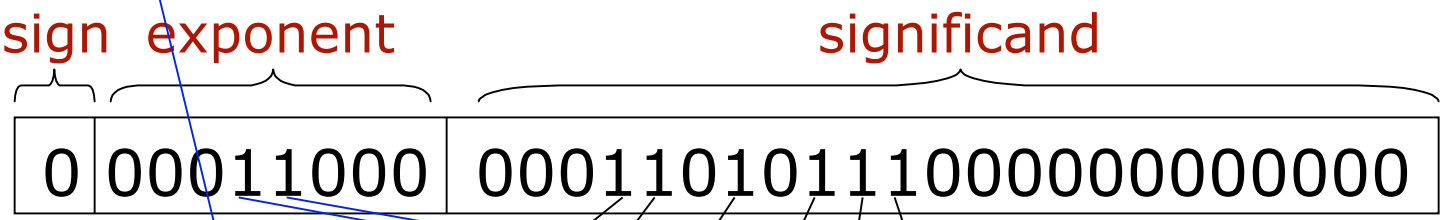
$= \quad 2^1 + 2^0 + 2^{-2} + 2^{-4} + 2^{-5} + 2^{-6} + 2^{-7}$

$= \quad 3\ {}^{47}/_{128}$

Floating-point number systems

$\pm\ (1\ .\ \text{significand})_2 \times\ 2^{\textbf{exponent}}$

Example:



$= \quad (1 + 2^{-4}+2^{-5}+2^{-7}+2^{-9}+2^{-10}+2^{-11}) \times\ 2^{\text{exponent}}$

$= \quad (1\ \mathbf{215/2048}) \times 2^{\text{exponent}}$

Floating-point number systems

Converting a number 284 into floating-point representation:

1. Determine the sign bit: 1 (negative), 0 (positive)
2. Represent the number in binary:
   100011100
3. Rewrite this value in "scientific notation", base 2 (with the binary point to the right of the most significant digit)
   1.00011100×2^8
4. The factional part of the above number is the significand:
   00011100000000000000000
5. If the exponent is 8 is represented by bit string 10000111
6. Final answer: 0 10000111 00011100000000000000000

Floating-point number systems

Converting a number 284 into floating-point representation:

1. Determine the sign bit: 1 (negative), 0 (positive)
2. Represent the number in binary:

   100011100

3. Rewrite this value in “scientific notation”, base 2 (with the binary point to the right of the most significant digit)

   1.00011100×2^8

4. The factional part of the above number is the significand:

   00011100

5. If exponent 8 is represented by bit string

Finite numeric value ranges

Every computer-number system has a **finite** value range:

- 32-bit unsigned integers range from **0** to **$2^{32}-1$**
- 32-bit signed integers range from **2^{-31}** to **$2^{31}-1$**

- 32-bit floating-point numbers range from **~ -2^{127}** to **~2^{127}**

Overflow errors

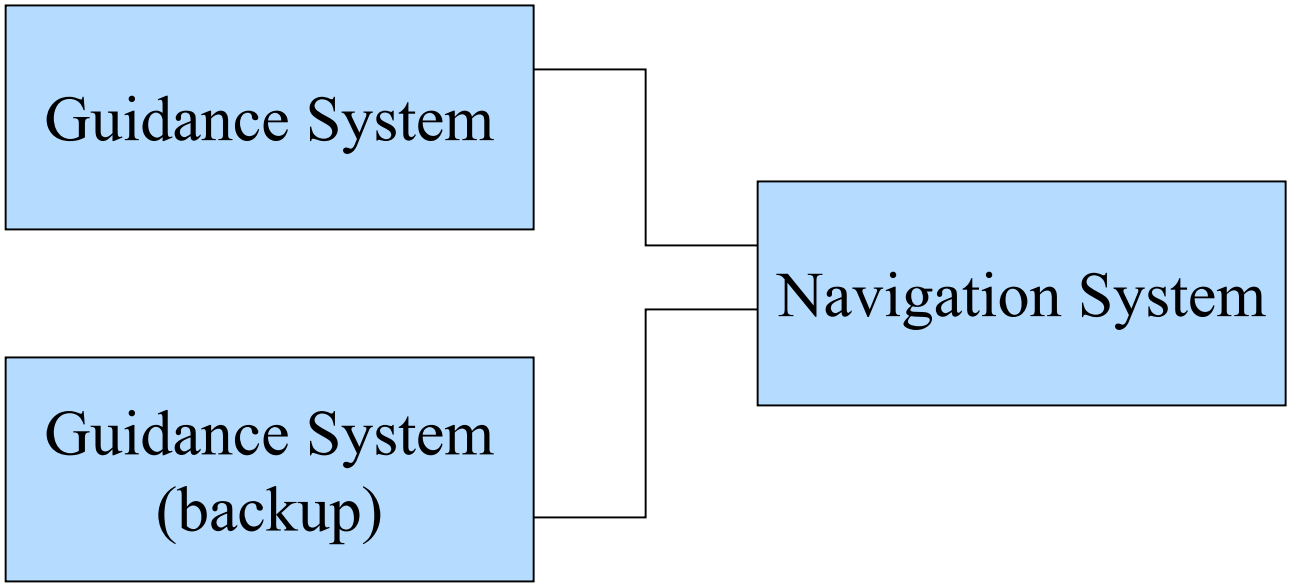
If two positive numbers are added together, their sum may exceed the largest value in the number system. In this case, we say an **overflow error** occurs.

Some programming languages and compilers issue a run-time error when an overflow has occurred.

Others simply discard the overflow bits and continue computing with the remainder value.

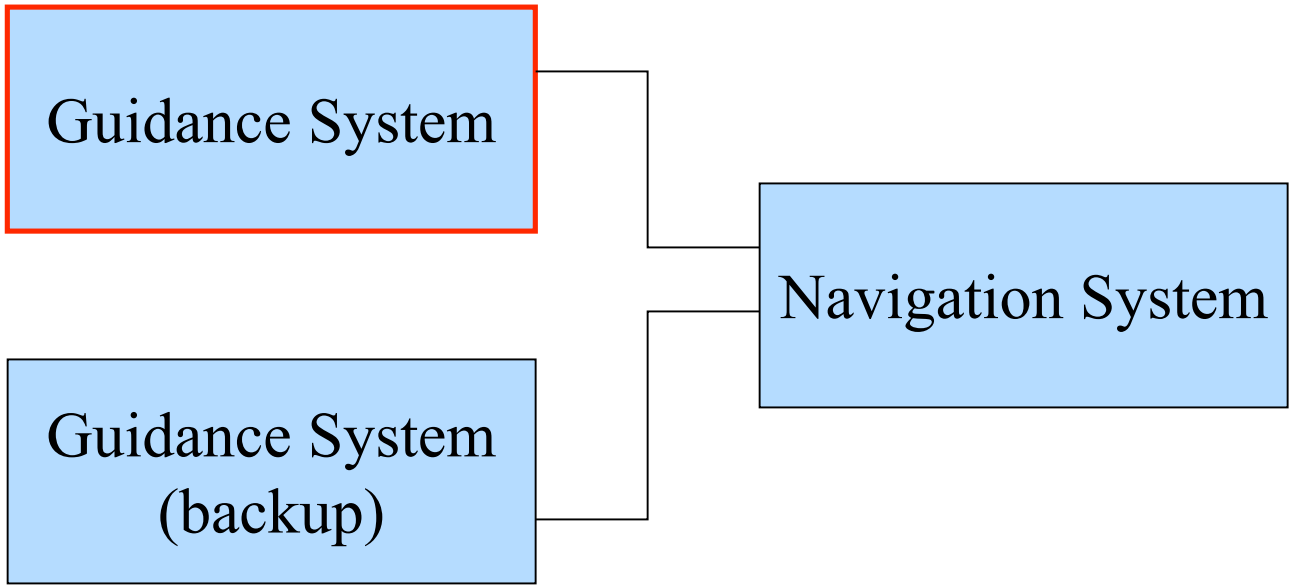
Ariane-5 software error

The Ariane-5 European Space Agency rocket self-destructed 40 seconds into its maiden flight.



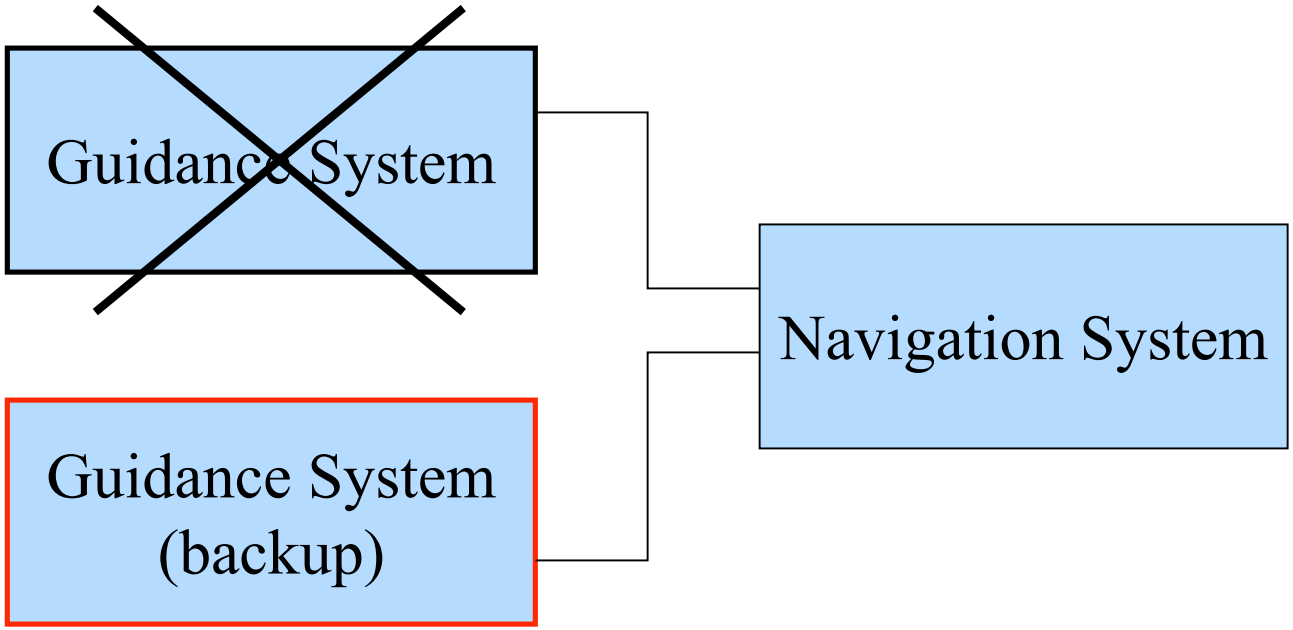
Ariane-5 software error

(1) The guidance system's computer tried to convert one piece of data — representing the rocket's sideways velocity — from a 64-bit format to a 16-bit format. The value was too big, and an overflow error occurred.



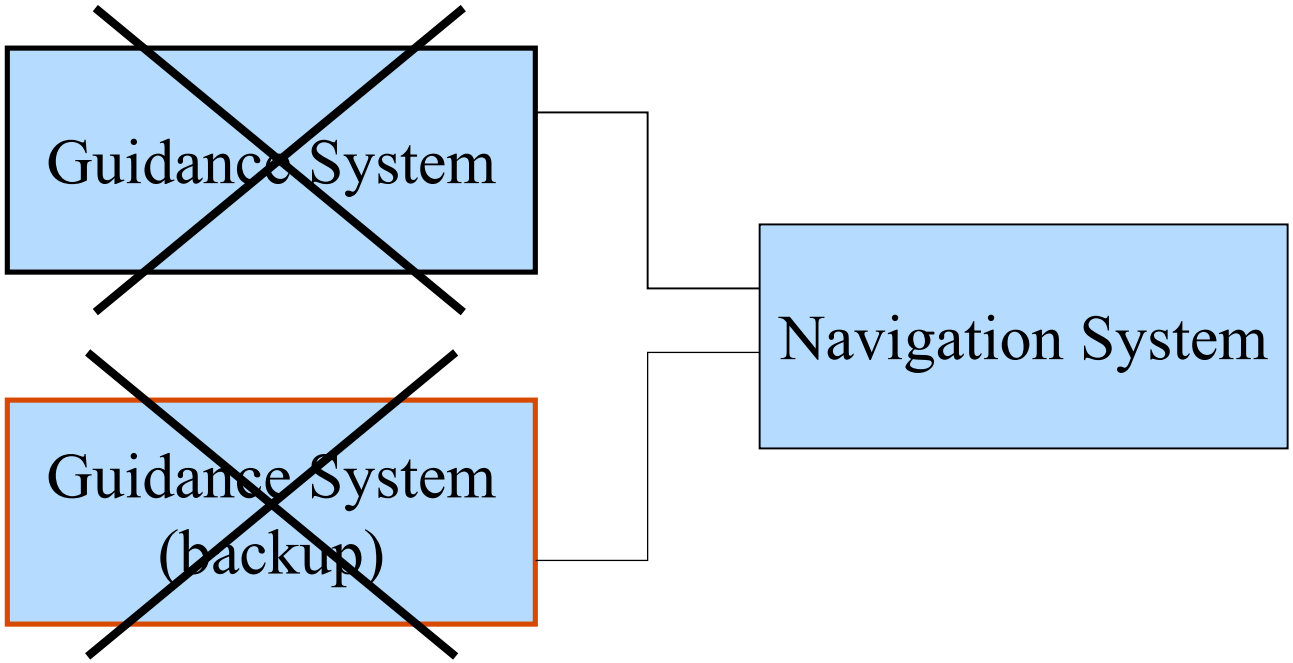
Ariane-5 software error

(2) The guidance system shuts down. Control passes to the backup guidance system.



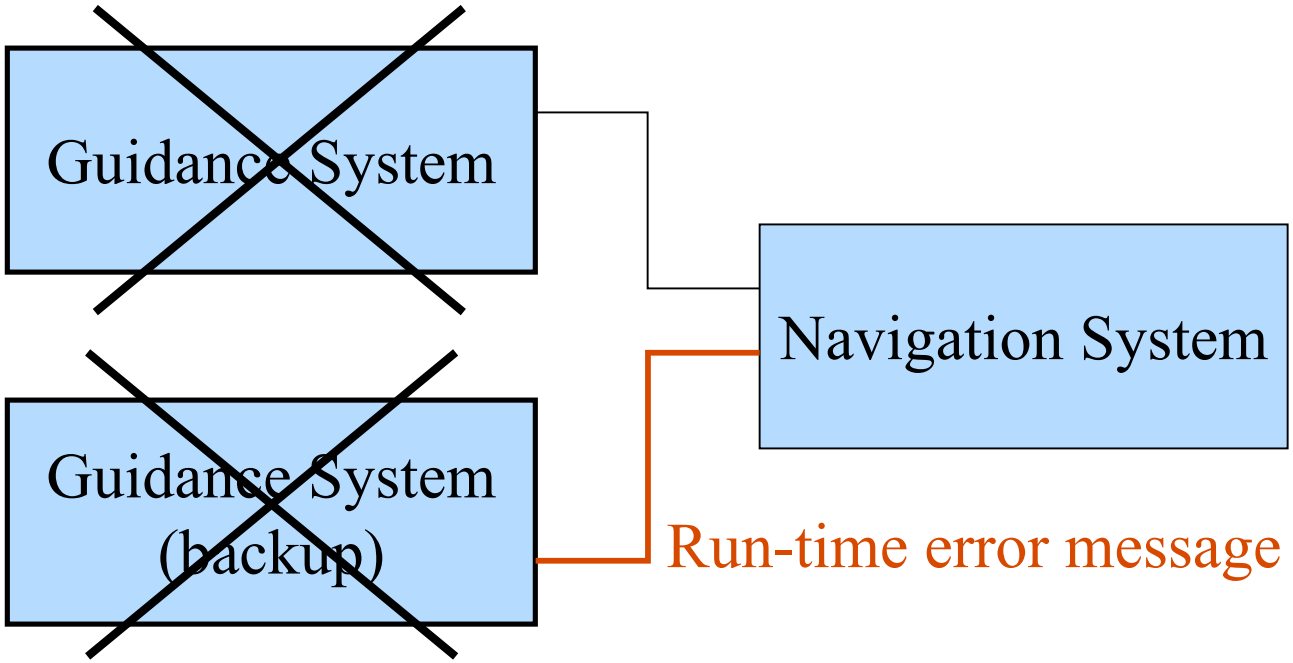
Ariane-5 software error

(3) The backup guidance system is running the same software. It shuts down in the same manner.



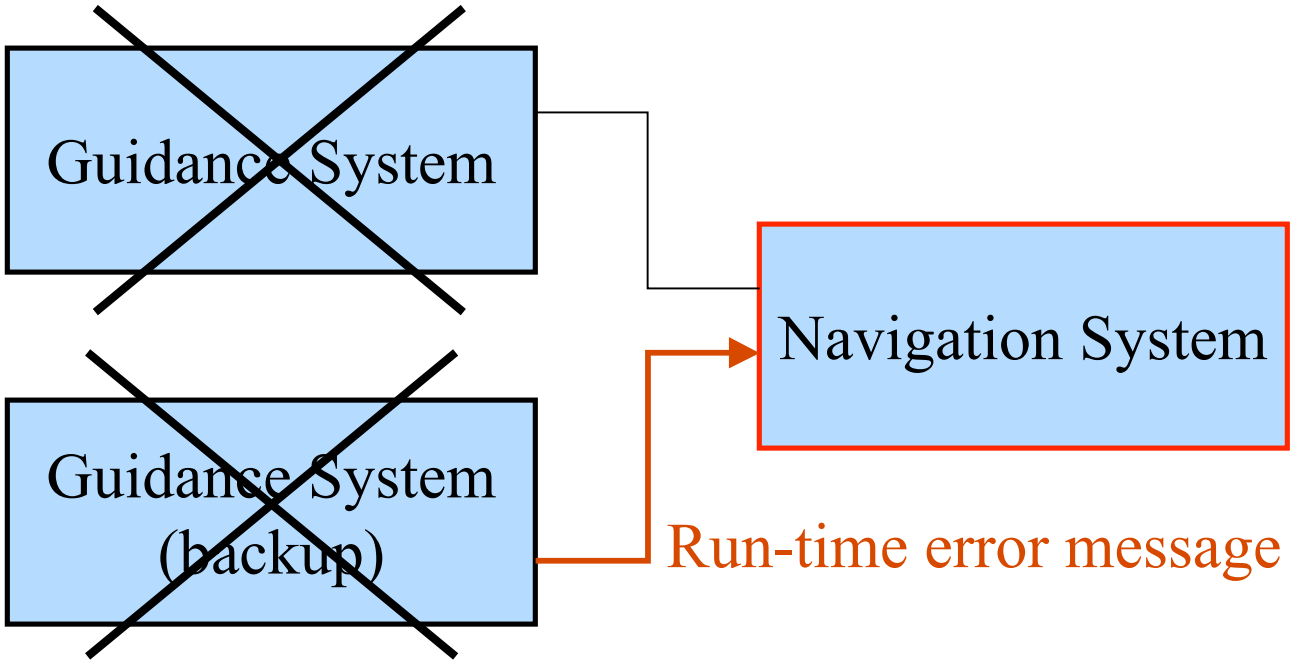
Ariane-5 software error

(4) The backup guidance system warns the navigation system that a run-time error has occurred.



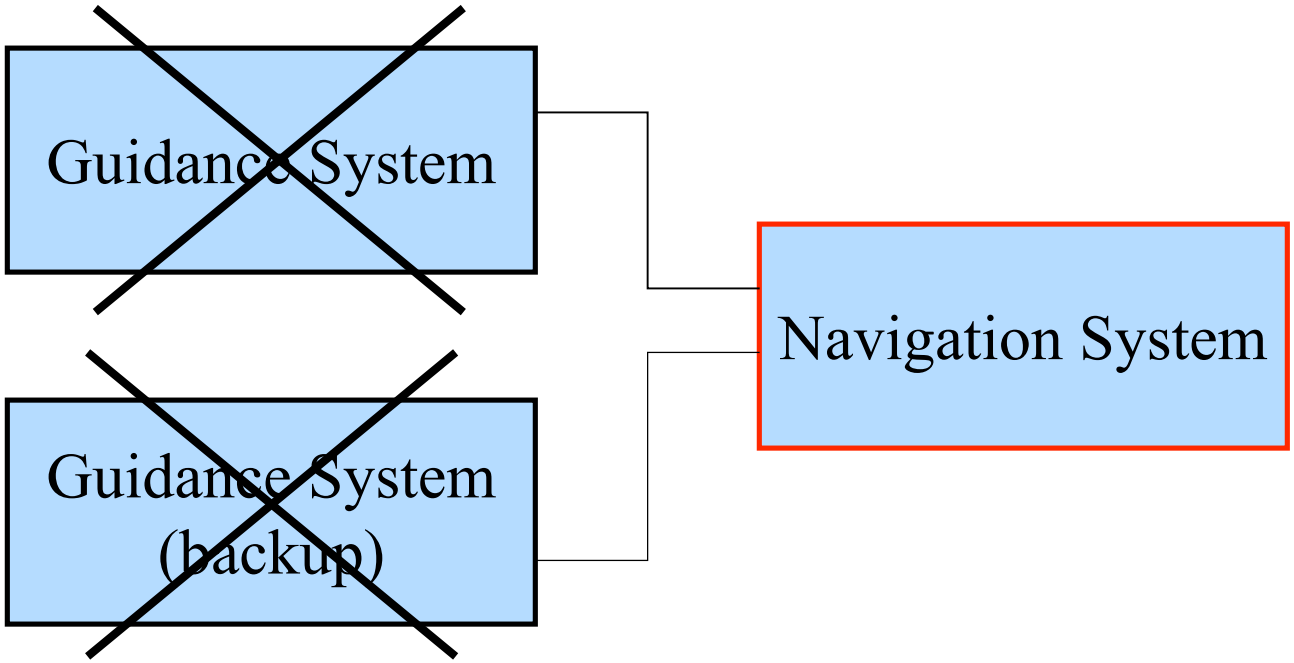
Ariane-5 software error

(5) The navigation interprets the error message as flight data — thinks that the rocket is horribly off course and attempts an abrupt course correction.



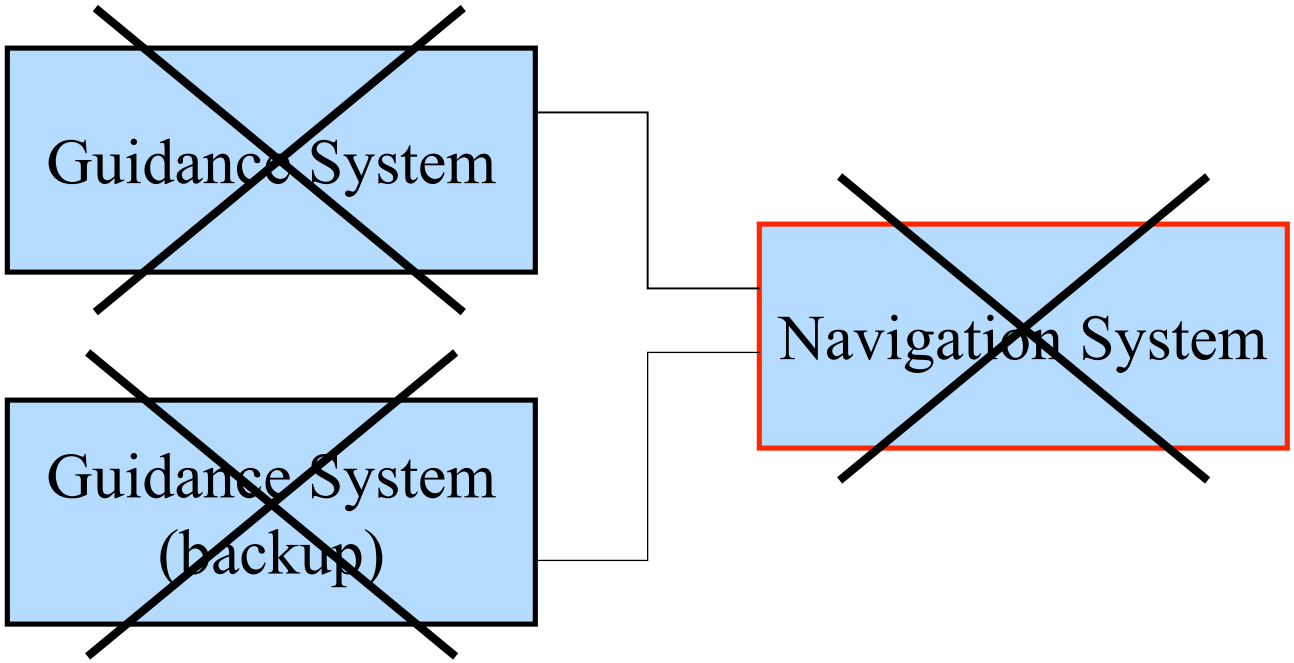
Ariane-5 software error

(6) The aerodynamic forces due to the course correction start tearing the boosters from the rocket

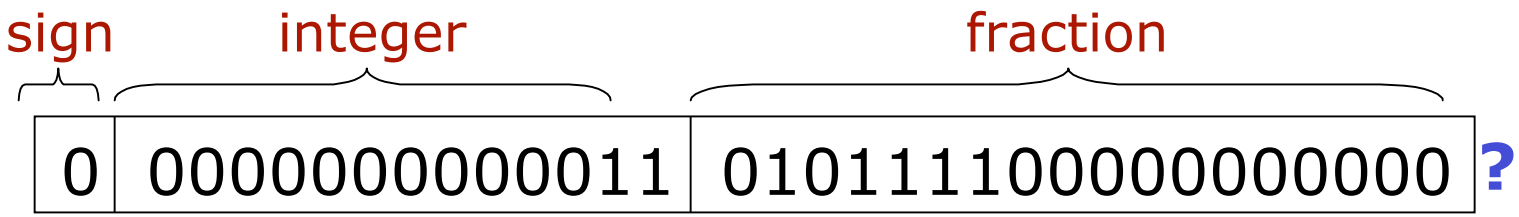


Ariane-5 software error

(7) Self destruction is automatically triggered less than 40 seconds into the flight.



Precision: Fixed-point



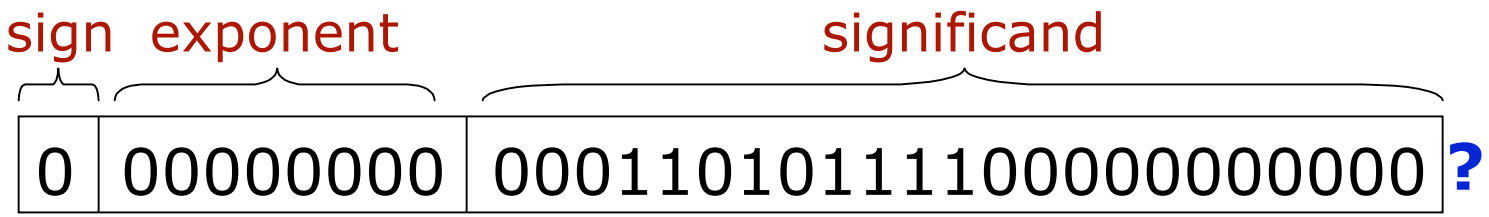
Limits the worst possible **absolute error**

Let x be a mathematical number, and x_{fixed} be the nearest fixed point number to x. Then

$$\left| x_{fixed} - x \right| \leq 2^{-19}$$

is the maximum possible absolute error

Precision: Floating-point



Limits the worst possible **relative error**

Let x be a mathematical number ($Sx2^E$), and x_{fl} be the nearest floating point number to x. Then

$|x_{fl} - x| \leq 2^{-19} \times 2^{E}$ where 2^{-18} is the machine epsilon ε

$|(x_{fl} - x) / x| \leq 2^{-19}$

What does this algorithm print out?

```
k=2;
oldsum = 0;
sum = 1;
while oldsum < sum do
    oldsum = sum;
    sum = sum + 1/k;
    k = k+1
end; /*while */
print k, sum;
```

IEEE standards for floating point

There are two IEEE[1] standardized floating point number systems that are broadly implemented.

- **Single precision** (32-bit word)

  uses 23 bits to represent significand

  $\varepsilon = 2^{-23} \cong 10^{-7}$

- **Double precision** (64-bit word)

  uses 52 bits to represent signficand

  $\varepsilon = 2^{-52} \cong 10^{-15}$

[1]Institute for Electrical and Electronics Engineers. Abbreviated, IEEE is pronounced "I triple E"

Precision

The U.S. federal budget is measured in trillions of dollars (~10^{12} dollars).

Near $x=10^{12}$,

- The single precision numbers are spaced about 10^5 (~ $100,000).

- The double precision numbers are 10^{-3} apart ($0.001).

Floating point operations

The IEEE standard requires that the result of a floating point operation be the rounded value of the exact (i.e., mathematical) result.

$$x \oplus y = \text{round}(x + y)$$
$$x \otimes y = \text{round}(x \times y)$$
$$x \oslash y = \text{round}(x / y)$$

IEEE allows several ways of rounding values. The default is that operation **round** rounds to the *nearest* floating point value.

Floating point operations

But the result of *two or more* arithmetic operations may not be the rounded value of the exact result.

Consider the computation

$$(x + y) - z,$$

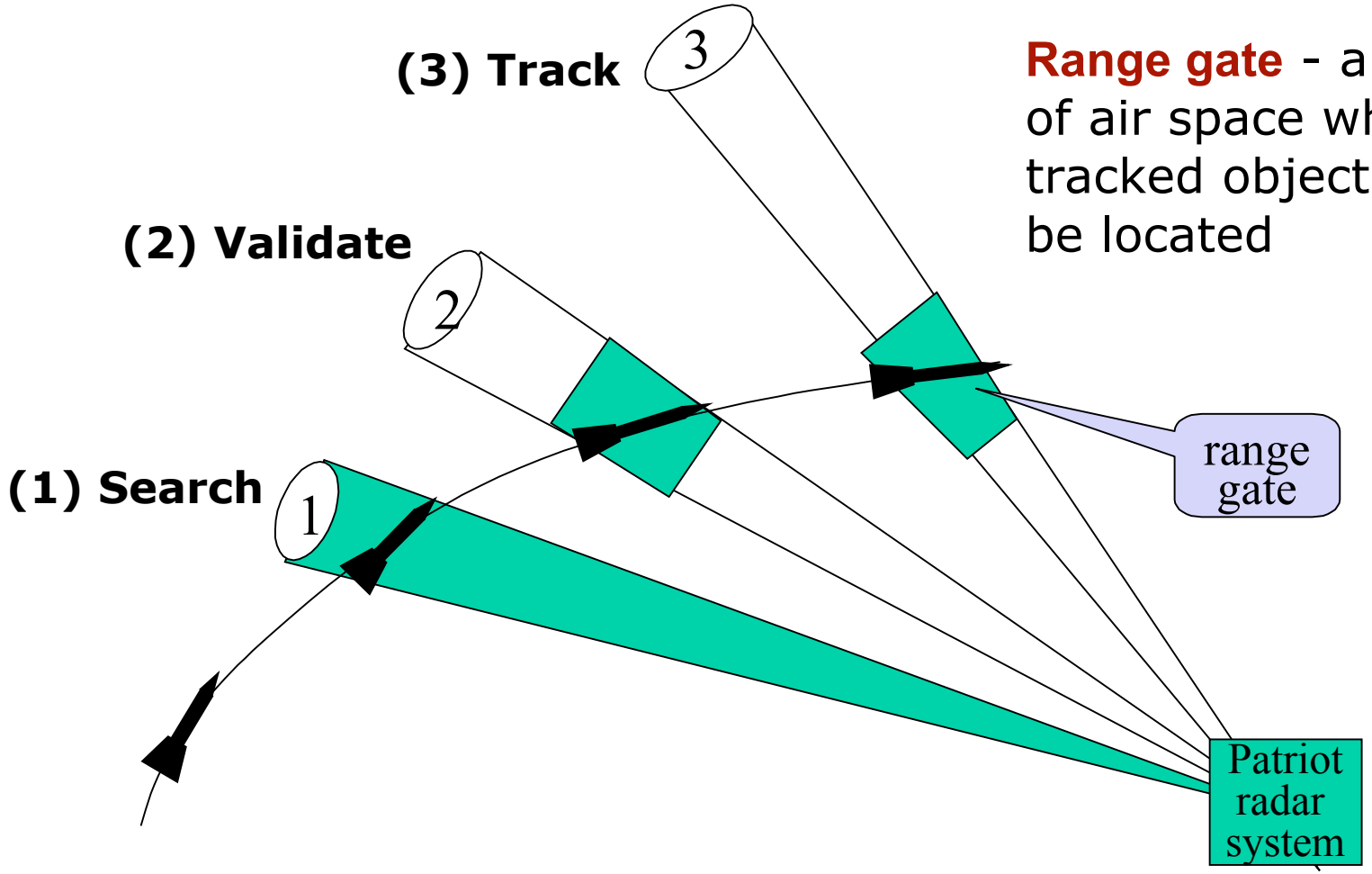
where $x=1$, $y=2^{-25}$, $z=1$.

Patriot missile software error

Patriot missile defense system was used heavily in the First Gulf War, to deflect Scud missile attacks on Israeli and U.S.-military targets.

The Patriot missiles were notoriously poor at hitting their targets - partially due to software arithmetic errors.

Patriot missile software



Range gate - area of air space where tracked object should be located

U.S. General Accounting Office, *GAO Report: Patriot Missile Defense*, Feb. 4, 1992

Patriot missile software error

The range gate's prediction of the Scud's next location is a function of the Scud's **known velocity** and the **time of last detection**.

Time is the number of tens of seconds since the last reboot, expressed as an integer. The longer the system has been running, the larger this number.

The time is multiplied by 1/10 to calculate time in seconds. The value 1/10 has a non-terminating binary value; was represented by 24-bit truncated value.

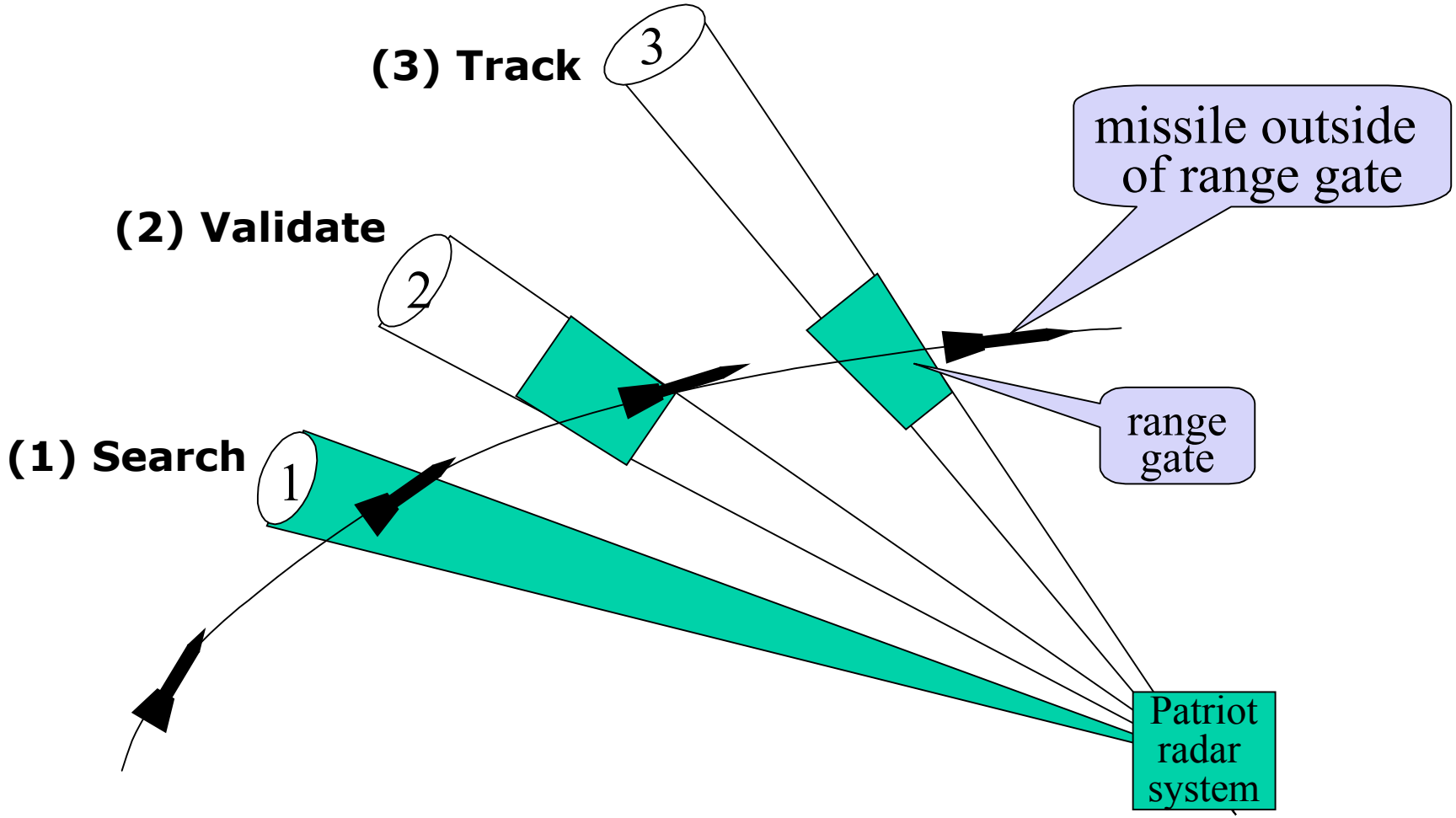
Patriot missile software error

On February 25, 1991, the patriot missile system in question had been operating for over **100 consecutive hours.**

Multiplying the truncated value of 1/10 by the number of tenths of seconds in 100 hours gives a time value that is off by **0.34 seconds.**

A Scud travels 1,676 m/s, and so travels **more than half a kilometer in this time.**

Patriot missile software error



Summary

Computer-based computations can be a **source** of error and uncertainty in calculated values.

- Overflow errors
- Inexact value representation
- Build-up of error from multiple floating-point operations

Quiz #1

- **In-lab quiz on Thursday October 7**
  - 45 min @ start of lab
  - 10% of your course mark
  - Closed book, closed notes
  - Math Faculty calculator allowed only

- **Old quizzes and explanations for some review answers are available on the SE101 web page**

Quiz #1

Covers

- Introduction to Engineering (IPE 1, lecture)
- Error propagation (IPE 10, 11,12 lectures)
- **NOT** Floating point numbers (Overton 3)

- Punctuation (Dupré 15,23,29,80,93,139)
- Sentence structure (Dupré 1,7,8,79,85,97)
- Report formatting (Dupré 21,26,96,126)

Quiz #1

Question #1: Explain to a non-technical person (e.g., a family member) what the discipline of software engineering is all about. Restrict your answer to 2-3 sentences.

Web Review #4

Will be available early Thursday, Oct. 7.

Will be due Tuesday, Oct. 12, 12:00 noon.

Readings for **next week's lecture and web review**:

Floating-point number system: Overton 3

Design Process IPE Ch. 15

Announcements

No office hours on Thursday October 7.