

# ارزیابی کارایی سیستم

## آقای دکتر مهدی امینیان

علیرضا سلطانی نشان

۲۹ آبان ۱۴۰۳

### فهرست مطالب

۲	۱ فصل اول: تعاریف اولیه
۲	۱.۱ ابزارها و تکنیک‌های ارزیابی سیستم
۲	۲.۱ متریک‌ها یا معیارها
۳	۳.۱ مخاطبان ارزیابی
۳	۴.۱ طراحی سیستم یا System design
۳	۵.۱ فضای System Selection
۳	۶.۱ تنظیم سیستم یا System tuning
۳	۷.۱ پیدا کردن مشکلات و گلوگاه‌ها Bottleneck identification
۳	۸.۱ حجم کاری یا Workload
۴	۹.۱ Capacity planning
۴	۱۰.۱ پیش‌بینی یا Forecasting
۴	۱۱.۱ الزامات انجام اندازه‌گیری‌های کارایی
۴	۱۲.۱ استفاده از تکنیک‌های آماری برای مقایسه جایگزین‌ها
۶	۱۳.۱ شبیه‌ساز و نکات تنظیمات
۶	۱۴.۱ هنر ارزیابی کارایی
۶	۱۵.۱ ارزیابی اصولی
۷	۱۶.۱ رویکرد سیستماتیک برای ارزیابی کارایی
۷	۱۷.۱ جنس متریک‌ها
۸	۱۸.۱ طراحی سناریو ارزیابی
۸	۱۹.۱ انتخاب تکنیک و متریک‌های ارزیابی
۸	۱.۱۹.۱ چه زمانی سریعاً به نتایج نیاز داریم؟
۸	۲.۱۹.۱ چه ابزارها و مهارت‌هایی موجود است؟
۸	۳.۱۹.۱ چه سطحی از دقت را برای ارزیابی باید در نظر گرفت؟
۹	۴.۱۹.۱ چه جایگزین‌هایی وجود دارد؟
۹	۵.۱۹.۱ قابلیت فروش یا Saleability
۹	۶.۱۹.۱ آیا می‌توانیم از دو یا چند تکنیک به صورت همزمان استفاده کنیم؟
۹	۲۰.۱ چگونه متریک‌های ارزیابی کارایی را انتخاب کنیم؟
۱۰	۱.۲۰.۱ چگونه معیارها را انتخاب کنیم؟ Best practice

## مجوز

به فایل license همراه این برگه توجه کنید. این برگه تحت مجوز GPLv۳ منتشر شده است که اجازه نشر و استفاده (کد و خروجی/pdf) را رایگان می‌دهد.

## کتاب مرجع

مرجع این جزوه تدریسی است که در کلاس صورت گرفته، و تدریس آن برگرفته از منابع زیر می‌باشد:

- Kishor S. Trivedi, Probability and Statistics with Reliability, Queuing, and Computer Science Applications, John Wiley and Sons, New York, 2001.
- Raj Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, John Wiley and Sons, Inc., New York, NY, 1991.
- Daniel A. Menasce, Virgilio A.F. Almeida, and Lawrence W. Dowdy, Performance by Design: Computer Capacity Planning by Example, Prentice Hall, 2004.
- K. Kant, Introduction to Computer System Performance Evaluation, McGraw-Hill Inc., 1992.
- Nader Nematollahi, Probability & Statistics for Engineers, IUST Press, 1382
- S. Mahmood Mirkhan, Stochastic Process, IAU Press, 1391

## ۱ فصل اول: تعاریف اولیه

### ۱.۱ ابزارها و تکنیک‌های ارزیابی سیستم

۱. مدل‌سازی ریاضیاتی (Modeling)

۲. اندازه‌گیری، در دنیای واقعی (Measurement)

۳. شبیه‌سازی (Simulation)

### ۲.۱ متریک‌ها یا معیارها

معیارهایی برای بررسی و ارزیابی عملکرد وجود دارد که بایستی در هنگام آنالیز و بررسی سیستم نسبت به آنها تنظیم شود. دقیقاً معیارها و شرایطی هستند که می‌خواهیم سیستم را بر مبنای آن اندازه‌گیری کنیم.

۱. Response time

۲. Throughput

۳. Energy consumption

۴. Memory consumption

### ۳.۱ مخاطبان ارزیابی

مخاطبانی که مورد ارزیابی کارایی سیستم‌ها قرار می‌گیرند معمولاً در دسته‌بندی زیر قرار می‌گیرند:

- برنامه‌نویسان (Developers and Programmers)
- طراحان سیستم (Designers)
- مدیران سیستم (System Administrators)
- کاربران (Users)

### ۴.۱ طراحی سیستم یا System design

تمامی افراد اکوسیستم یک نرم‌افزار، سخت‌افزار یا شبکه در مسئله ارزیابی کارایی سیستم درگیر خواهند بود. هدف اصلی در ارزیابی سیستم ارائه بالاترین کارایی با کمترین هزینه می‌باشد. مهندسی علمی برای ایجاد توازن بین کارایی و هزینه‌ها می‌باشد. هزینه‌ها شامل زمان، نیروی انسانی و پول می‌باشند. هیچ وقت ادعای نظری برتری کارایی یک چیز را ثابت نمی‌کند.

### ۵.۱ فضای System Selection

فضای Selection بحث در انواع سیستم‌ها را دارد. چیزی که کمک می‌کند تا بهترین انتخاب را داشته باشیم با بررسی ارزیابی کارایی آن سیستم.

یکی از مهم‌ترین کاربردهای ارزیابی کارایی سیستم، آنالیز (تجزیه و تحلیل) می‌باشد. یک سیستمی که تمام فرآیند توسعه را طی کرده باشد و نیاز به بررسی رخدادهای آن دارد که مشخص شود چرا در این هنگام باعث کاهش کارایی آن شده است. اگر کد و آزمون‌های سیستمی به خوبی در هنگام توسعه و آزمایش انجام شده باشد هزینه‌های پشتیبانی در آینده کاهش می‌یابد. ارزیابی کارایی کاربرد بسیار زیادی در جنبه‌های انتخاب سیستم در لیستی از انتخاب‌ها، طراحی سیستم‌ها و اپلیکیشن‌ها و آنالیز سیستم‌های کنونی دارد.

### ۶.۱ تنظیم سیستم یا System tuning

وقتی که یک سیستم را با مفاهیم ارزیابی کارایی تنظیم می‌کنیم که بتوانیم بهترین کارایی را برای سیستم خود طراحی و اجرا کنیم.

### ۷.۱ پیدا کردن مشکلات و گلوگاه‌ها Bottleneck identification

پیدا کردن مشکلات و گلوگاه‌هایی در سیستم که منجر به کاهش کارایی سیستم مورد نظر شده است. این فاز کلاً به صورت شناسایی مشکلات سیستم کنونی معرفی می‌شود.

### ۸.۱ حجم کاری یا Workload

تمام ویژگی‌هایی که بار کاری در سیستم ایجاد می‌کند. تا زمانی که به سیستم ورودی داده نشود نمی‌توانیم بررسی کنیم که خروجی به چه شکلی خواهد بود و آیا می‌توانیم از خروجی اندازه‌گیری کنیم؟ پس به ورودی داده‌ها Workload می‌گوییم. این که قرار است چه نوع داده‌ای به سیستم وارد شود، تعداد مخاطب سیستم چقدر است و تمامی جنبه‌هایی که بر روی اندازه‌گیری کارایی سیستم تاثیرگذار است را در این

بخش بررسی می‌کنیم. باید بدانیم که سیستم مورد ارزیابی برای چه نوع ورودی‌هایی تنظیم شده و تمام متریک‌های دیگر روی عملکرد آن تاثیرگذار است را شناسایی کنیم.

به عبارتی دیگر، درخواست‌هایی که توسط کاربران یا برنامه‌های دیگر جهت استفاده از سیستم مورد نظر وارد می‌شود را Workload می‌گوییم.

## ۹.۱ Capacity planning

تعیین تعداد و اندازه کامپوننت‌ها، ماژول‌ها و بخش‌های استفاده شده در سیستم را گوییم. ظرفیت‌های سیستم را بررسی می‌کنیم تا بدانیم برای آن چقدر منابع باید اختصاص دهیم. در حقیقت از دسته فعالیت‌هایی برای آینده سیستم می‌باشد. برای مثال یک نرم‌افزار قرار است پنج سال در دسترس کاربران باشد، باید در آن به صورت پویا مشخص کنیم که کاربران چقدر افزایش خواهند داشت و داده‌های آن‌ها با منابعی که مستقر کردیم سازگاری ظرفیتی دارد یا بایستی برنامه‌ریزی مجدد روی منابع جدید نسبت به افزایش درخواست‌ها، داشته باشیم. بیشتر با مقیاس پذیری همراه است.

## ۱۰.۱ پیشبینی یا Forecasting

پیشبینی کارایی سیستم در لودهای کاری آینده. همانند برنامه‌ریزی برای ظرفیت‌ها می‌باشد. بیشتر روی لودهایی که در آینده روی عملکرد سیستم تاثیرگذار است همراه می‌باشد.

## ۱۱.۱ الزامات انجام اندازه‌گیری‌های کارایی

برای اندازه‌گیری کارایی یک سیستم بایستی دو بخش را داشته باشیم:

۱. ایجاد داده و بار کاری یا Load generator

۲. پایش وضعیت سیستم با بار کاری‌ای که ایجاد شده یا Monitor

برای مثال موارد ارزیابی لیست شده زیر را در نظر بگیرید و بگویید که کدام ابزار می‌تواند برای سنجش کارایی مناسب باشد:

- میزان مصرف یک LAN که معمولاً با استفاده از نرم‌افزار شبیه‌سازی توان مصرفی iPerf قابل سنجش می‌باشد.
- مدت زمانی که طول می‌کشد یک پاسخ از Web server در سمت سرویس‌گیرنده دریافت شود.

- Python Locust: جهت شبیه‌سازی ارسال درخواست‌های تعداد زیاد در برنامه‌های Backend و Api‌ها

- Apache benchmark: تنها برای اندازه‌گیری میزان تحمل بار درخواست در سرویس‌های راه‌اندازی شده Apache مورد استفاده قرار می‌گیرد.

- JMeter

- بررسی کیفیت صدا در شبکه‌های VoIP

## ۱۲.۱ استفاده از تکنیک‌های آماری برای مقایسه جایگزین‌ها

وقتی در یک System selection چندین سیستم را داریم بایستی مقایسه‌ای از نظر عملکردی بر روی آن‌ها انجام شود که بدانیم کدام سیستم به نیاز ما نزدیک‌تر می‌باشد. برای اینکار در نظر گرفتن چند نکته الزامی می‌باشد:

۱. اصولاً فرایند ارزیابی با یک ورودی یا Workload انجام نمی‌شود بلکه در حالت‌های مختلف با ورودی‌های مختلف بایستی سیستم را مورد ارزیابی قرار داد. با اینکار ممکن است به رخدادهای غیرقطعی برسیم که می‌تواند روی عملکرد سیستم تاثیرگذار باشد.

۲. هر کدام از متریک‌ها می‌تواند از فرمول ریاضیاتی خاص خودش استفاده کند و نتیجه را ارائه دهد. می‌تواند یکی از متریک‌ها با میانگین‌گیری باشد و دیگری با بدست آوردن واریانس.

یکی از قسمت‌های مهم ارزیابی استفاده از حالت‌های مختلف ورودی‌ها جهت مقایسه کارایی سیستم می‌باشد. فرآیند ارزیابی بایستی بارها امتحان شود تا بتوان به نتیجه‌گیری مناسبی رسید. از یک یا چند روش آماری جهت جمع‌بندی کارهایمان در مقالات و آزمایش‌ها بایستی استفاده کنیم تا دلیل برتری کارایی سیستم (الف) از سیستم (ب) مشخص شود. جدول شماره ۱۲.۱ تعداد بسته‌های گم شده را در لینک‌های (A) و (B) را نشان می‌دهد.

File size	Link A	Link B
۱۰۰۰	۵	۱۰
۱۲۰۰	۷	۳
۱۳۰۰	۳	۰
۵	۰	۱

Table : ۱ A و B لینک‌های در بسته‌ها شدن گم مقایسه

سوال مهمی که این میان ممکن است مطرح شود آن است، آزمایش‌های اندازه‌گیری و شبیه‌سازی چه تعداد بار بایستی انجام شود که قادر به پوشش فضای مورد نظر باشد؟

غالباً فاکتورهای بسیار زیادی روی کارایی تاثیر می‌گذارند، بایستی اثراتی که به صورت انفرادی مهم هستند را جدا کنیم. تعدادی از پارامترها به صورت ثابت یا Static هستند. برای مثال در سناریوهای خود نوع سیستم عامل و مرورگری که کاربر استفاده می‌کند را به صورت ثابت به ترتیب ویندوز و کروم در نظر می‌گیریم. یا به عنوان مثالی دیگر، ممکن است در یک پروژه تعداد متنوعی از CPUها را داشته باشیم اما روی همه سیستم‌ها تنها یک سیستم عامل مشخص (لینوکس) را نصب می‌کنیم. یا به عنوان مثالی دیگر، اندازه بسته‌هایی که در شبکه ارسال می‌کنیم همگی 3KB می‌باشد اما تعداد ارتباطات شبکه و درخواست‌هایی که باید پاسخ داده شود کاملاً پویا می‌باشد. دلیل اصلی این کار آن است که تمام پارامترهایی که سیستم می‌پذیرد را نمی‌توان کامل ارزیابی کنیم چرا که از نظر هزینه‌ای مدیریت دشوار و پیچیده‌ای دارد. راهکار اصلی آن است که بر اساس تجربه یکسری از پارامترها را حذف می‌کنیم.

قابل توجه است که همه پارامترها در ابتدا برای ارزیابی کارایی سیستم بایستی نوشته شوند و به صورت هوشمندانه تعدادی از این پارامترها را ثابت می‌کنیم.

- پارامترهایی که به صورت ثابت نگهداری می‌شوند را Static parameter می‌گوییم.
- پارامترهایی دستخوش تغییرات قرار می‌گیرند را Dynamic parameter یا Factor parameter می‌گوییم.

## نکات

- عملیات ارزیابی کارایی بسیار دشوار و زمان‌بر می‌باشد.
- سطح تعدادی از پارامترها را می‌توان زیاد کرد.
- فرایند ارزیابی کارایی به صورت iterative و تکرارشونده هستند.
- همیشه باید کل سیستم را بررسی و آنالیز کنیم که پارامترهای ثابت و factor مشخص شود.
- عملیات ارزیابی آنقدر تکرار می‌شود که بدانیم سیستم روی چه پارامترهایی باعث می‌شود که ارزیابی را با دقت بررسی کنیم.
- انجام آزمایش‌ها براساس نتایج و سناریوها را Experiment می‌گوییم.

## ۱۳.۱ شبیه‌ساز و نکات تنظیمات

یکی از نکات مهم در ارزیابی کارایی سیستم انجام شبیه‌سازی درست و مناسب می‌باشد:

- انتخاب زبان درست، انتخاب Seed برای اعداد تصادفی، مدت زمان اجرای شبیه‌سازی و آنالیزها
- قبل از موارد بالا بایستی اعتبار شبیه‌ساز تایید شود و شبیه‌ساز کاملاً با معیارهای ارزیابی ما بایستی Tune شود.
- برای مثال، جهت مقایسه کارایی دو الگوریتم کش جایگزینی خواهیم داشت:
- به چه اندازه‌ای شبیه‌سازی باید در حال اجرا باشد؟
- چه کاری باید انجام دهیم که با زمان اجرای کوتاه‌تر به دقت اجرای بلند مدت برسیم؟
- استفاده از مدل‌های صف‌بندی برای ارزیابی کارایی
- غالباً می‌توان سیستم‌های کامپیوتری را با نرخ سرویس‌ها و بارکاری‌ای که به سیستم می‌رسد به دو روش مدل کرد:
  - استفاده از چند سرویس‌دهنده یا Multiple servers
  - استفاده از چند صف‌بندی یا Multiple queues
  - برای مثال می‌گوییم که کدام روش می‌تواند باعث ارزیابی بهتر کارایی وب سرویس می‌شود، استفاده از دو تک‌پردازنده یا استفاده از چهار وب سرویس تک پردازنده؟

## ۱۴.۱ هنر ارزیابی کارایی

ارزیابی را نمی‌توان به صورت مکانیکی تولید کرد. نیازمندی اصلی ارزیابی، دانش دقیق نسبت به سیستم و دقت بالا در انتخاب متدولوژی، Workload ها و ابزارها را دارا می‌باشد. با تکنیک‌ها و ابزارهایی که در اختیار داریم باید بررسی کنیم که با استفاده از آن‌ها به چه شکلی می‌توانیم به ارزیابی کارایی سیستم برسیم. پروژه‌ها اکثراً شامل مستندات Test coverage می‌باشد که در آن تکلیف دو طرف یعنی هم کارفرما هم پیمانکار را مشخص می‌کند.

## ۱۵.۱ ارزیابی اصولی

نکات مهمی در مورد ارزیابی اصولی به صورت Best practice ها مطرح می‌شود:

- یک هدف واحد را در نظر بگیریم.
- در فرایند ارزیابی کارایی تمام اهداف در ابتدا طراحی و مشخص شده‌اند و اصلاً اهداف به صورت کلی و جامع نیستند.
- هدف مغرضانه یا Biased goals نداشته باشیم.
- هیچ وقت از سیستمی طرفداری و جانبداری بی‌دلیل نکنیم و براساس داده‌سازی‌هایی که کرده‌ایم نشان ندهیم که سیستم تولید شده توسط ما بهتر از سیستم مقابل کارایی دارد. باید منصفانه تمام ارزیابی‌ها انجام شود.
- تمام Workload هایی که در سیستم مورد نظر استفاده می‌شود بایستی جامع و کامل باشد نه اینکه به صورت خاص تعیین شوند که برای سیستم طراحی شده توسط ما کارایی خوبی داشته باشد و برای سیستم مورد مقایسه کارایی بد. تنها با داده‌های خیلی زیاد و یا خیلی کم ارزیابی را انجام ندهیم بلکه تمام Workload ها بایستی از سطح مناسبی از تنوع (Diversity) برخوردار باشند.
- تکنیک‌های اشتباه ارزیابی: از مدل، شبیه‌سازی و ابزار اندازه‌گیری مناسبی استفاده کنیم.
- از یک سطح مناسب و متعادلی از جزئیات استفاده کنیم.

- نمی‌توانیم تعداد زیادی از موارد را مانند مثال سیستم عامل‌ها و مرورگرها را برای سیستم خود مورد ارزیابی قرار دهیم.
- نمی‌توانیم تعداد کمی از موارد را انتخاب کنیم.
- نداشتن حساسیت روی آنالیزها
- آن ارزیابی که بر روی سیستم انجام می‌شود تنها یک مشاهده می‌باشد نه یک دامنه یا fact طبیعی که بتوانیم با ادعاهایی سیستم را زیر سوال ببریم و اثبات کارمان را انجام دهیم.
- نمایش درستی از آنچه که بدست آورده‌ایم را فراهم کنیم.
- خواندن صرف برای همه کار پیچیده‌ای است، سعی کنیم با نمایش نمودارها و گراف‌ها نتایج را ملموس‌تر سازیم.
- در استفاده از شکل‌ها نهایت احتیاط را داشته باشیم. اگر نتایج به صورت نامناسب مصورسازی شده باشد باعث می‌شود ارزیابی سیستم زیر سوال رود.
- استفاده مناسب از محدودیت‌ها و مفروضات
- احتمال دارد در قدم اول فرض انبوه ترافیک شبکه را در TCP داشته باشیم با اینکه بیشتر ترافیک از سمت UDP ناشی می‌شود.
- لزومی ندارد که مفروضات در سیستم مورد نظر همیشه درست باشند.

## ۱۶.۱ رویکرد سیستماتیک برای ارزیابی کارایی

۱. بیان اهداف و تعریف کردن مرزها
۲. انتخاب متریک‌های مناسب ارزیابی
۳. لیست کردن پارامترها و Workload های سیستمی
۴. انتخاب فاکتورهای پویا و ثابت
۵. انتخاب تکنیک مناسب ارزیابی
۶. انتخاب Workload
۷. طراحی سناریو ارزیابی
۸. آنالیز و شبیه‌سازی و تفسیر کردن داده‌ها
- (آ) نموداری که برای نمایش نتایج استفاده می‌کنیم کارایی خطی را نشان می‌دهد یا نمایی؟
۹. نمایش مناسب نتایج
۱۰. تکرار در هر مرحله در صورت لزوم

## ۱۷.۱ جنس متریک‌ها

متریک‌های بررسی کارایی معمولاً از سه جنس زیر هستند:

- سرعت: میزان گذردهی و تاخیر
- دقت: نرخ خطاهای سیستم
- میزان در دسترس بودن: ارسال و دریافت موفق بسته‌ها

## ۱۸.۱ طراحی سناریو ارزیابی

برای داشتن بهترین نتیجه با تلاش کم دو فاز زیر را می‌توانیم در طراحی سناریو در نظر داشته باشیم:

۱. فاز اول:

- (آ) انتخاب فاکتورهای زیاد اما با سطح پایین
- (ب) بررسی اینکه کدام فاکتور حائز اهمیت است.

۲. فاز دوم:

- (آ) انتخاب فاکتورهای کم اما با سطوح بیشتر
- (ب) پیدا کردن محدوده‌ای که بیشترین تاثیر را روی فاکتورها دارد.

## ۱۹.۱ انتخاب تکنیک و متریک‌های ارزیابی

بررسی اینکه در چه لایف سایکل از سیستم هستیم؟ در چه مرحله‌ای از توسعه نرم‌افزار هستیم؟ اگر سیستم مورد نظر به طور کامل پیاده‌سازی شده باشد استفاده از ابزارهای اندازه‌گیری انتخاب مناسبی هستند.

### ۱.۱۹.۱ چه زمانی سریعاً به نتایج نیاز داریم؟

استفاده از Modeling ریاضی برای تکنیک ارزیابی کارایی سیستم نسبت به شبیه‌سازی و اندازه‌گیری سریع‌تر می‌باشد.

### ۲.۱۹.۱ چه ابزارها و مهارت‌هایی موجود است؟

۱. زبان برنامه‌نویسی که باید برای شبیه‌سازی برنامه مورد نظر یاد بگیریم. کاملاً وابسته به زبان پیاده‌سازی هستیم.

(آ) استفاده از MatLab

(ب) استفاده از Python

(ج) استفاده از Opnet

۲. ابزارهایی برای پشتیبانی از اندازه‌گیری مانند

(آ) Packet sniffers

(ب) Source codes

۳. مهارت در بررسی مدل‌ها مانند تئوری صف (Queueing theory)

(آ) همه چیز به صورت دانشی است که به وسیله ریاضی آن را نمایش می‌دهیم.

(ب) Analytic modeling

### ۳.۱۹.۱ چه سطحی از دقت را برای ارزیابی باید در نظر گرفت؟

مدل‌سازی اصطلاحاً Course grade یعنی جواب‌ها و خروجی‌هایی که دارد کاملاً به صورت جامع می‌باشد که نقطه مقابل آن اندازه‌گیری می‌باشد که با جزئیات نتایج را به صورت عمیق بررسی می‌کند.  
سطوح بیان جزئیات از سطح بالا به سمت عمیق:

۱. Math modeling



### ۴.۱۹.۱ چه جایگزین‌هایی وجود دارد؟

بحث دیگری که در بررسی و ارزیابی کارایی سیستم مورد استفاده قرار می‌گیرد استفاده از جایگزین‌ها می‌باشد. به معنای آن است که نتایج بدست آمده از فرایند ارزیابی را می‌خواهیم بارها تکرار کنیم تا به یک سنجش و Trade off بین انتخاب‌های موجود برسیم، نکته آن است که مدل‌سازی تکرار پذیری راحت‌تری دارد اما در مقابل اندازه‌گیری یا Measurement تکرار در رسیدن به خروجی شامل هزینه بالا می‌باشد. یعنی اثر مستقیمی در بررسی هزینه‌ها دارد.

### ۵.۱۹.۱ قابلیت فروش یا Saleability

بین تمام صحبت‌هایی که انجام شد، وقتی می‌خواهیم یک فرایند ارزیابی را انجام دهیم، باید ببینیم کارفرما (مشتري نهایی) چه تکنیکی را برای محصول خودش می‌خواهد. حالت عمومی بین تمام این مشتری‌ها استفاده از تکنیک اندازه‌گیری یا Measurement می‌باشد. اما در بحث‌های آکادمیک استفاده از مدلینگ ریاضیاتی و شبیه‌سازی استفاده بیشتری دارد.

### ۶.۱۹.۱ آیا می‌توانیم از دو یا چند تکنیک به صورت همزمان استفاده کنیم؟

امکان‌پذیر می‌باشد اما بایستی نکات زیر را در نظر بگیریم:

۱. یک تکنیک، تکنیک دیگر را تایید کند.

۲. اکثر مقالات با کیفیت، ارزیابی کارایی محصول یا کار مورد نظر خود را با اندازه‌گیری و شبیه‌سازی انجام می‌دهند.

ما بایستی نسبت به نیازمندی پروژه خود برنامه‌ریزی ارزیابی را مشخص کنیم که از چه تکنیک یا تکنیک‌هایی بایستی استفاده کنیم. شکل شماره ۶.۱۹.۱.

معیارها	مدل‌سازی	شبیه‌سازی	اندازه‌گیری
مراحل	فرقی ندارد	فرقی ندارد	از مرحله پروتوتایپ
زمان	کم	متوسط	متفاوت و زیاد
نوع ابزار	تحلیل‌گرانه	دیداری و ترکیبی	ابزارهای دقیق
دقت	کم	متوسط	متفاوت در عمل
Trade-off	راحت	متوسط	سخت
هزینه	اندک	متوسط	زیاد
قابلیت فروش	اندک	متوسط	زیاد

جدول ۲: مقایسه بین تکنیک‌های ارزیابی

### ۲۰.۱ چگونه متریک‌های ارزیابی کارایی را انتخاب کنیم؟

اندازه‌گیری متریک‌های یک سیستم جهت ارزیابی آن بایستی به صورت مشخص شده باشد در غیر این صورت از نظر هزینه زمانی و مالی بسیار طولانی خواهد شد. نکته: دلیل جداسازی و تشکیل خانواده بین متریک‌های ارزیابی سیستم این است که بتوانیم استانداردهای قابلیت اطمینان را جداگانه انجام دهیم یا به عبارتی دیگر استفاده از روتین‌های آزمایش متفاوت می‌باشد. هزینه‌ها، نگاه‌ها و روتین‌ها کاملاً متفاوت می‌باشد.

متریک‌ها از سه خانواده هستند:

۱. Speed

(آ) Time

(ب) Rate

(ج) Resource

۲. Reliability

(آ) Probability

(ب) Time between

۳. Availability

(آ) Duration

(ب) Time between

### ۱.۲۰.۱ چگونه معیارها را انتخاب کنیم؟ Best practice

۱. تغییرپذیری پایین به حدی که کمترین تکرار را داشته باشد.

۲. تا آنجایی که امکان دارد متریک‌های redundant را حذف کنیم که پیچیدگی ارزیابی را زیاد نکند.

۳. متریکی انتخاب کنیم که جامع و کامل باشد تا ابعاد بزرگی از سیستم را پوشش دهد.

### ۲.۲۰.۱ متریک‌هایی که به صورت عمومی استفاده می‌شوند

● Response time

– Turn around time: زمان بین ارسال یک کار، تکمیل و رسیدن به خروجی مورد نظر

– Reaction time: زمان بین ارسال یک درخواست و شروع اجرای آن

– Stretch factor: مدت زمانی که طول می‌کشد که سیستم پاسخ مورد نظر را ارسال کند.

\* بیشتر سیستم‌ها زمانی که بار کاری آن‌ها افزایش پیدا می‌کند زمان پاسخ‌دهی آن‌ها نیز افزایش می‌یابد.

● Throughput

– Operations/seconds

\* Batch: Request Per Second

\* Interactive: Job Per Second

\* MIPS: Millions of Instructions Per Second

\* MFLOPS: Millions of Floating-Point Ops Per Second

\* Networks: Packets Per Seconds یا Bits Per Second

\* Transactions Processing TPS: Transactions Per Second

– Capacity

– Efficiency

\* برای مثال بررسی بهینه بودن مصرف باتری در دو یا چند رویکرد مطرح شده بر اساس ارسال اطلاعات در ثانیه.

Utilization -

- دسته‌بندی‌های Utility:

\* HB: Higher is better برای مثال Throughput

\* LB: Lower is better برای مثال مدت زمان پاسخ به درخواست‌ها

\* Nominal is best: برای مثال Utilization

● Reliability: اتکاپذیری و قابلیت اعتماد به سیستم زمانی که برنامه نسبت به هر خطایی مقاوم است و هر خطایی باعث شکنندگی برنامه نخواهد شد.

Uptime -

MTTF -