==<font color="#800000">'''关键词'''</font>==

<font color="#808080">

AWS & Compute

</font>

==<font color="#800000">'''逻辑模型'''</font>==

<font color="#808080">

</font>

==<font color="#800000">'''学习报告正文'''</font>==

<font color="#808080">

===<font color="#800000">'''课程目标'''</font>===

===<font color="#800000">'''课程内容'''</font>===

\*这节课感觉收获了很多，的确比写代码要有趣得多。

\*感觉上完课接着进行学习报告课程回顾的部分效果会更好。

===='''1.顾老师部分'''====

\*首先顾老师向我们介绍了Mathematica和NetLogo中一个重要的概念就是Time，在MMA中叫Dynamic（动态），在Netlogo里Loop。使得这两个软件里都有了时间的概念。虽然我暂时还不知道这意味着什么。

\*其次顾老师又一次地强调了这门课的核心是计算思维，是一种思维，而不是教我们去编程，不是让我们去当码农，不是教让我们去刷题。虽然刷题在基础的构建上有很大的作用，但是我们不能只是去刷题。

\*工程领域不能随便说创新，而更多的是一种策略，因为工程行业的很多东西是容不得我们去乱创新的，一旦这样整个计算机网络的体系都可能崩坏掉。对于这点我的理解是可以看作如果汇编改变它的语法，可能在局部上使其更简单，但是汇编以上的整个庞大的计算机体系就要全部崩坏。但是在最后与顾老师的对话中，顾老师说我们要在整个体系构建完成之后才进行所谓创新，就是说，把一件事的整体的基本的构架做出来之后再去改进，不能你说什么就是什么。也就是说，我们需要在一定的基础上通过策略（战略性调整）完善我们的体系（抽象的或是具体的）。

\*对于计算思维的，还有一点就是在构造出一个产品之后我们需要用自然语言编写出一个说明书性质的东西来连接我们和产品，同时吸引更多的人加入产品的论证过程，这个说明说越简练清晰越好，因为这样更容易让一个从未接触过这个产品的人迅速形成对这个产品的整体构架，从而使得对产品的论证变得可能。（举例：Linux OS，有一个比较完善的wiki系统充当着说明书的角色，使得全世界的人都来完善这个操作系统）

\*在得到一个问题，我们首先要做的不是去想最底层怎么改进能更快，而是将整个体系搭建完成，构建出一套完整的解决这个问题的体系再去分配各项任务，从而形成一个任务流以及监督体系。

\*怎么优化计算机。就是把计算机的东西都藏起来，从最早的电脑到现在，只要敲敲屏幕，说一句话，就能做想做的事，藏起来了很多东西。

===='''2.ASW & Compute'''====

\*首先我们要有自己的思维方式，要有自由意识，要去收集自己感兴趣方面的信息，要有自己的一个小方向，不能缺失理想。（例子：招收实习生时，谈及自己的兴趣，很多人说的是让我干什么我就干什么）否则就成了一个没有追求的码农。

\*还是创新，不能天马星空，首先要有自己坚实的基础。不能打破计算机的最基本合约。

\*云计算（cloud computing）是基于互联网的相关服务的增加、使用和交付模式，通常涉及通过互联网来提供动态易扩展且经常是虚拟化的资源。云是网络、互联网的一种比喻说法。过去在图中往往用云来表示电信网，后来也用来表示互联网和底层基础设施的抽象。因此，云计算甚至可以让你体验每秒10万亿次的运算能力，拥有这么强大的计算能力可以模拟核爆炸、预测气候变化和市场发展趋势。用户通过电脑、笔记本、手机等方式接入数据中心，按自己的需求进行运算。

\*对云计算的定义有多种说法。对于到底什么是云计算，至少可以找到100种解释。现阶段广为接受的是美国国家标准与技术研究院（NIST）定义：云计算是一种按使用量付费的模式，这种模式提供可用的、便捷的、按需的网络访问， 进入可配置的计算资源共享池（资源包括网络，服务器，存储，应用软件，服务），这些资源能够被快速提供，只需投入很少的管理工作，或与服务供应商进行很少的交互。

====='''ASW'''=====

\*AWS即Amazon Web Services，是亚马逊（Amazon）公司的云计算IaaS和PaaS平台服务

\*'''ASW的分类'''

计算类：

#EC2（Elastic Compute Cloud）是一种弹性云计算服务，可为用户提供弹性可变的计算容量，通常用户可以创建和管理多个虚拟机，在虚拟机上部署自己的业务，虚拟机的计算能力（CPU、内存等）可以根据业务需求随时调整。

#Elastic IP Addresses（弹性IP地址） – 弹性IP地址是为动态云计算设计的静态IP地址。一个弹性IP地址是和你的账户相关，而不是和你的一个特定实例相关。不像传统的静态IP地址，弹性IP地址可以通过重新匹配你的共有IP地址到你账户任意的实例，从而让你可以忽略实例或者可用区域的错误。连接本质上是通过NAT1:1的匹配每个Elastic IP和Private IP。

#Elastic MapReduce：EMR采用运行在亚马逊EC2和S3的托管Hadoop框架上。以立即获得满足需要的计算能力，例如网页索引、数据挖掘等数据密集型任务，轻松、经济地处理海量数据，不用担心对Hadoop集群耗时的设置、管理或调优。

#AS（Auto Scaling）自动伸缩服务：允许用户根据需要控制亚马逊EC2自动扩大或减小计算能力。用户利用AS可以无缝地增加EC2的实例数量，以保证使用高峰期的性能，也可以在需求停滞时自动减少以降低成本。AS特别适合那些需求按小时、天或周规律变化的应用程序。AS由亚马逊CloudWatch控制，并且用户不必支付CloudWatch以外的其他服务费用。

#ELB (Elastic Load Balancing)弹性负载平衡：自动将入口流量分配到多个亚马逊EC2实例上。弹性负载平衡在实例池中不断检测不正常的实例，并自动引导路由流量到正常的实例上，直到不正常的实例恢复正常。客户可以在单一的数据中心进行负载平衡，更可以在跨中心的应用上获得相同的功能。兼容IPv6，数据来自于CloudWatch

部署&管理类：

#ACW (Amazon CloudWatch)云监控服务：监控亚马逊自身提供的云资源以及在云上运行的应用程序。提供可视化监测，并且可以利用API调用进一步处理监控的数据。

#Amazon WorkSpaces：是一种虚拟桌面服务，托管在Amazon的云中。用户可以选择任何终端设备（如笔记本电脑、iPad、Kindle Fire或Android平板电脑）访问 Amazon WorkSpaces，获得与传统办公桌面一样的使用体验，更能享受节约设备成本、保证个人数据安全、随时随地办公等便利。

网络类：

#R53（Amazon Route 53）亚马逊53号路由：Domain Name System web service(网络域名服务)。提供从基础设施（EC2实例，ELB，或者S3）到IP地址的映射。

#VPC (Virtual Private Cloud)虚拟私有云：在亚马逊公有云之上创建一个私有的，隔离的云。可以像在自己的数据中心一样定义VPC的拓扑结构。可以和公司现有的数据中心互通。可以利用NAT使得子网不暴漏内网IP，公用一个IP地址与外界通讯。通过NAT设置访问控制，保护数据安全性。

存储类：

#S3 (Simple Storage Service) ：亚马逊简单存储服务（S3）是一种网络存储服务，可为用户提供持久性、高可用性的存储。用户可以将本地存储迁移到Amazon S3，利用 Amazon S3 的扩展性和按使用付费的优势，应对业务规模扩大而增加的存储需求，使可伸缩的网络计算更易于开发。

#EBS (Elastic Block Store)弹性数据块存储：EBS卷是独立于实例的存储，可作为一个设备动态连接到运行着的亚马逊EC2实例上。EBS特别适合于单独需要一个数据库、文件系统、或访问原始块存储的应用程序。

应用服务类：

#SQS (Simple Queue Service)简单消息队列服务：提供消息存储队列，使消息可以在计算机之间传递，在执行不同任务的分布式应用组件之间轻松的转移数据，既不会丢失信息，也不要求每个组件都保持可用。SQS可以与亚马逊EC2和其他AWS的基础设施网络服务紧密结合在一起，方便地建立自动化的工作流程。SQS以网络服务的形式运行，对外发布一个web消息框架。Internet中任何计算机都可以添加或阅读消息，而不必安装任何软件或配置特殊的防火墙。使用SQS的应用组件可以独立运行，不需要在同一网络中使用相同的技术开发，也不必在同一时间运行。

#SNS (Simple Notification Service)简单通知服务：在云中安装、处理或发送通知。它为开发人员提供了一种从应用程序发布消息，并立即传送给订阅者或其他应用程序的能力，用于创建通知某应用程序（或客户）某方面的主题。客户订阅这些主题，并使用客户选定的通信协议（例如，HTTP，电子邮件等）发布消息。亚马逊SNS的潜在用途包括监控，工作流系统，时间敏感的信息更新，移动应用等等。

数据库类：

#SDB (Amazon SimpleDB)简单数据库：非关系型数据存储服务

#RDS (Relational Database Service)：是一种基于云的关系型数据库服务，用户可以在云中配置、操作和扩展关系数据库。Amazon RDS 支持 MySQL、Oracle、Microsoft SQL Server 或 PostgreSQL 等关系型数据库。用户无需本地维护数据库，由Amazon RDS为用户管理。

支付类：

#FPS (Flexible Payments Service)灵活支付服务

#ADP (Amazon DevPay)亚马逊支付设计

内容交付类：

#CloudFront 云前：整合亚马逊其他云服务产品，完成高效快速的分布式内容交互。

人工服务类：

#AMT (Amazon Mechanical Turk)机械的土耳其人：“机械的土耳其人”一词来源：这个名字源自于臭名远扬的能下象棋的“自动装置”，它是匈牙利男爵沃尔夫冈·冯·肯佩伦（Wolfgang Von Kempelen）1770年建造的。这个木制机器外形像一个坐在大机箱前的土耳其魔法师，它能自动而快速地下象棋，用复杂的齿轮和杠杆系统来移动棋子。在维也纳皇宫的首次表演中，它就迅速击败了对手Cobenzl伯爵，让在场的皇室成员看得十分高兴。从此关于这个惊人聪明的机器人迅速闻名于世，于是肯佩伦带着它在欧洲各地表演，击败了一系列著名的挑战者，包括拿破仑和本杰明·富兰克林。直到几年之后，这个骗局才被揭穿。原来机箱里藏了一名象棋大师，他用一个磁铁系统来跟踪对手的举动并移动自己的棋子，这个人实际是在模拟一种人工智能。

===='''HPC'''====

HPC是高性能计算(High Performance Computing)机群的简称。指能够执行一般个人电脑无法处理的大资料量与高速运算的电脑，其基本组成组件与个人电脑的概念无太大差异，但规格与性能则强大许多。现有的超级计算机运算速度大都可以达到每秒一兆（万亿，非百万）次以上。“超级计算”（supercomputing）这名词第一次出现，是在1929年《纽约世界报》关于IBM为哥伦比亚大学建造大型报表基（tabulator）的报导。

===='''其他概念'''====

\*AZ：

\*数据中心：

\*挖矿：

===='''个人回顾【比较杂乱】'''====

森林老师【抱歉忘记叫啥。。。】向我们举了很多关于云计算的功用，上面已经列出了许多，

\*可以用很少的成本获得计算性能极高的计算资源。（500强中第76名）

\*EC2是老师主要讲的，其中涉及到计算资源的分配（商用问题）。

\*讲到数据中心时提到了容灾的重要性，数据中心必须保持随时运转，（就像飞机不能中途停下来一样），对于损坏的硬盘需要及时更换（可以自动化吗？）。

\*云计算，我们只需要传我们需要计算的数据上去，返回值也只是一些数据，所以可以很快。

\*我们只需要写Code传上去，无关乎系统。

===='''云计算'''====

\*分布式计算（Distribute Compute）

\*\*分布式计算是一种计算方法，和集中式计算是相对的。随着计算技术的发展，有些应用需要非常巨大的计算能力才能完成，如果采用集中式计算，需要耗费相当长的时间来完成。分布式计算将该应用分解成许多小的部分，分配给多台计算机进行处理。这样可以节约整体计算时间，大大提高计算效率。

\*并行计算

\*\*并行计算或称平行计算是相对于串行计算来说的。它是一种一次可执行多个指令的算法，目的是提高计算速度，及通过扩大问题求解规模，解决大型而复杂的计算问题。所谓并行计算可分为时间上的并行和空间上的并行。 时间上的并行就是指流水线技术，而空间上的并行则是指用多个处理器并发的执行计算。

\*虚拟化

\*\*虚拟化，是指通过虚拟化技术将一台计算机虚拟为多台逻辑计算机。在一台计算机上同时运行多个逻辑计算机，每个逻辑计算机可运行不同的操作系统，并且应用程序都可以在相互独立的空间内运行而互不影响，从而显著提高计算机的工作效率。

\*\*虚拟化使用软件的方法重新定义划分IT资源，可以实现IT资源的动态分配、灵活调度、跨域共享，提高IT资源利用率，使IT资源能够真正成为社会基础设施，服务于各行各业中灵活多变的应用需求。

\*负载均衡

\*\*负载均衡 建立在现有网络结构之上，它提供了一种廉价有效透明的方法扩展网络设备和服务器的带宽、增加吞吐量、加强网络数据处理能力、提高网络的灵活性和可用性。

\*\*负载均衡，英文名称为Load Balance，其意思就是分摊到多个操作单元上进行执行，例如Web服务器、FTP服务器、企业关键应用服务器和其它关键任务服务器等，从而共同完成工作任务。

\*冗余

\*\*在通信工程当中，冗余指出于系统安全和可靠性等方面的考虑，人为地对一些关键部件或功能进行重复的配置。当系统发生故障时，比如某一设备发生损坏，冗余配置的部件可以作为备援，及时介入并承担故障部件的工作，由此减少系统的故障时间。冗余尤用于应急处理。冗余可以存在于不同层面，如网络冗余、服务器冗余、磁盘冗余、数据冗余等。

===='''Docker'''====

Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的容器中，然后发布到任何流行的 Linux 机器上，也可以实现虚拟化。容器是完全使用沙箱机制，相互之间不会有任何接口。[http://baike.baidu.com/link?url=tW1bcIvdMxJiMil1nCyHOWdI\_uCcenMoUP4TZ6PZU2CTe4n3lROfRpuoxl5EMNIEnoKuipwonI\_rnn4J0g1rZq][https://en.wikipedia.org/wiki/Docker\_(software)]

===='''ASW Lambda'''====

AWS Lambda是一个用于部署代码、管理服务以及监控轻量级服务运行状态的细粒度方法。

[http://baike.baidu.com/link?url=-f1MXuXqPwGvi-AgDwdliqL7fzZA6fOl--fpP3YaakRKDNJbU-iusmNyqyI1ud2MfM2GUUyYuRg3rj\_6Ys648fh7QMS3aN3gPGNO1BirY0q][https://en.wikipedia.org/wiki/AWS\_Lambda]

===<font color="#800000">'''课程回顾'''</font>===

===<font color="#800000">'''学习感悟'''</font>===

</font>

===<font color="#800000">'''对课程整体构架的理解'''</font>===

===<font color="#800000">'''学习报告阅读'''</font>===

===<font color="#800000">'''课程建议'''</font>===

==<font color="#800000">'''编辑词条'''</font>==

==<font color="#800000">'''创建词条'''</font>==

==<font color="#800000">'''参考文献'''</font>==

\*[https://aws.amazon.com/cn/?sc\_channel=PS&sc\_campaign=acquisition\_CN&sc\_publisher=baidu&sc\_medium=brandzone&sc\_content=pc&sc\_detail=title&sc\_category=pc&sc\_segment=101&sc\_matchtype=exact&sc\_country=CN| AWS官网]

\*[http://baike.baidu.com/link?url=tW1bcIvdMxJiMil1nCyHOWdI\_uCcenMoUP4TZ6PZU2CTe4n3lROfRpuoxl5EMNIEnoKuipwonI\_rnn4J0g1rZq| Docker]

\*[http://baike.baidu.com/link?url=-f1MXuXqPwGvi-AgDwdliqL7fzZA6fOl--fpP3YaakRKDNJbU-iusmNyqyI1ud2MfM2GUUyYuRg3rj\_6Ys648fh7QMS3aN3gPGNO1BirY0q| AWS Lambda]

\*[http://baike.baidu.com/link?url=\_-iFkI1mxSP1MMXySImD0iwsxoOV\_OvNXYbBetO\_EKd2gtAY\_OpX\_wKJjs8tgqBglG6JoHlZVE1jklIYYZS1oaFqIU8NNXpLt0wA12Wg18QX4\_kc\_UAGKpyhI6kL-QPE| 云计算]

\*[http://baike.baidu.com/item/%E5%88%86%E5%B8%83%E5%BC%8F%E8%AE%A1%E7%AE%97/85448| 分布式计算]

\*[http://baike.baidu.com/view/1666.htm| 并行计算]

\*[http://baike.baidu.com/view/729629.htm| 虚拟化]

\*[http://baike.baidu.com/view/51184.htm| 负载均衡]

\*[http://baike.baidu.com/subview/104445/10102122.htm#viewPageContent| 冗余]

\*[http://baike.baidu.com/subview/290722/15933139.htm#viewPageContent| HPC]

==版权声明==

[[File:BYNCND.png | link=http://creativecommons.org/licenses/by-nc-nd/3.0/cn/ ]] [http://creativecommons.org/licenses/by-nc-nd/3.0/cn/ 署名-非商业性使用-禁止演绎 3.0 中国大陆 (CC BY-NC-ND 3.0 CN)]