

# Logic and Programming in Logic

Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Introduction: Logic and Logic Programming Languages. Introduction: Logic and Logic Programming Languages.

Logic, according to Webster's dictionary [Webster, 1987], is 'a science that deals with the principles and criteria of validity of inference and demonstration: the science of the formal principles of reasoning.' Such 'principles and criteria' are always described in terms of a language in which inference, demonstration, and reasoning may be expressed. One of the most useful accomplishments of logic for mathematics is the design of a particular formal language, the First Order Predicate Calculus (FOPC). FOPC is so successful at expressing the assertions arising in mathematical discourse that mathematicians and computer scientists often identify logic with classical logic expressed in FOPC. In order to explore a range of possible uses of logic in the design of programming languages, we discard the conventional identification of logic with FOPC, and formalize a general schema for a variety of logical systems, based on the dictionary meaning of the word. Then, we show how logic programming languages may be designed systematically for any sufficiently effective logic, and explain how to view Prolog, Datalog, ?Prolog, Equational Logic Programming, and similar programming languages, as instances of the general schema of logic programming. Other generalizations of logic programming have been proposed independently by Meseguer [Meseguer, 1989], Miller, Nadathur, Pfenning and Scedrov [Miller et al., 1991], Goguen and Burstall [Goguen and Burstall, 1992]. The purpose of this chapter is to introduce a set of basic concepts for understanding logic programming, not in terms of its historical development, but in a systematic way based on retrospective insights. In order to achieve a systematic treatment, we need to review a number of elementary definitions from logic and theoretical computer science and adapt them to the needs of logic programming. The result is a slightly modified logical notation, which should be recognizable to those who know the traditional notation. Conventional logical notation is also extended to new and analogous concepts, designed to make the similarities and

differences between logical relations and computational relations as transparent as possible. Computational notation is revised radically to make it look similar to logical notation.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Equational Logic Programming. Equational Logic Programming. Sections 2.3.4 and 2.3.5 of the chapter ‘Introduction: Logic and Logic Programming Languages’ are crucial prerequisites to this chapter. I summarize their relevance below, but do not repeat their content. Logic programming languages in general are those that compute by deriving semantic consequences of given formulae in order to answer questions. In equational logic programming languages, the formulae are all equations expressing postulated properties of certain functions, and the questions ask for equivalent normal forms for given terms. Section 2.3.4 of the ‘Introduction . . .’ chapter gives definitions of the models of equational logic, the semantic consequence relation  $T \models (t_1 \approx t_2) \dots (t_1 \approx t_2 \text{ is a semantic consequence of the set } T \text{ of equations, see Definition 2.3.14), and the question answering relation } \dots$  (norm  $t_1, \dots, t_i : t$ )  $\approx$   $(t \approx s) \dots (t \approx s \text{ asserts the equality of } t \text{ to the normal form } s, \text{ which contains no instances of } t_1, \dots, t_i, \text{ see Definition 2.3.16}).$

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Meta-Programming in Logic Programming. Meta-Programming in Logic Programming.

A meta-program, regardless of the nature of the programming language, is a program whose data denotes another (object) program. The importance of meta-programming can be gauged from its large number of applications. These include compilers, interpreters, program analysers, and program transformers. Furthermore, if the object program is a logic or functional program formalizing some knowledge, then the meta-program may be regarded as a meta-reasoner for reasoning about this knowledge. In this chapter, the meta-program is assumed to be a logic program. The object program does not have to be a logic program although much of the work in this chapter assumes this. We have identified three major topics for consideration. These are the theoretical foundations of meta-programming, the suitability of the alternative meta-programming techniques for different applications, and methods for improving the efficiency of meta-programs. As with logic programs generally, meta-programs have declarative and procedural semantics. The theoretical study of meta-

programming shows that both aspects of the semantics depend crucially on the manner in which object programs are represented as data in a meta-program. The second theme of the paper is the problem of designing and choosing appropriate ways of specifying important meta-programming problems, including dynamic metaprogramming and problems involving self-application. The third theme concerns efficient implementation of meta-programs. Meta-programming systems require representations with facilities that minimize the overhead of interpreting the object program. In addition, efficiency can be gained by transforming the meta-program, specializing it for the particular object program it is reasoning about. This chapter, which concentrates on these aspects of meta-programming, is not intended to be a survey of the field. A more complete survey of meta-programming for logic programming can be found in [Barklund, 1995]. Many issues in meta-programming have their roots in problems in logic which have been studied for several decades. This chapter emphasizes meta-programming solutions. It is not intended to give a full treatment of the underlying logical problems, though we try to indicate some connections to wider topics in meta-logic. The meta-programs in this chapter are logic programs based on first order logic.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Higher-Order Logic Programming. Higher-Order Logic Programming.

Modern programming languages such as Lisp, Scheme and ML permit procedures to be encapsulated within data in such a way that they can subsequently be retrieved and used to guide computations. The languages that provide this kind of an ability are usually based on the functional programming paradigm, and the procedures that can be encapsulated in them correspond to functions. The objects that are encapsulated are, therefore, of higher-order type and so also are the functions that manipulate them. For this reason, these languages are said to allow for higher-order programming. This form of programming is popular among the users of these languages and its theory is well developed. The success of this style of encapsulation in functional programming makes it natural to ask if similar ideas can be supported within the logic programming setting. Noting that procedures are implemented by predicates in logic programming, higher-order programming in this setting would correspond to mechanisms for encapsulating predicate expressions

within terms and for later retrieving and invoking such stored predicates. At least some devices supporting such an ability have been seen to be useful in practice. Attempts have therefore been made to integrate such features into Prolog (see, for example, [Warren, 1982]), and many existing implementations of Prolog provide for some aspects of higher-order programming. These attempts, however, are unsatisfactory in two respects. First, they have relied on the use of ad hoc mechanisms that are at variance with the declarative foundations of logic programming. Second, they have largely imported the notion of higher-order programming as it is understood within functional programming and have not examined a notion that is intrinsic to logic programming. In this chapter, we develop the idea of higher-order logic programming by utilizing a higher-order logic as the basis for computing. There are, of course, many choices for the higher-order logic that might be used in such a study. If the desire is only to emulate the higher-order features found in functional programming languages, it is possible to adopt a “minimalist” approach, i.e., to consider extending the logic of first-order Horn clauses—the logical basis of Prolog—in as small a way as possible to realize the additional functionality.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Proof Procedures for Logic Programming. Proof Procedures for Logic Programming.

A proof procedure is an algorithm (technically, a semi-decision procedure) which identifies a formula as valid (or unsatisfiable) when appropriate, and may not terminate when the formula is invalid (satisfiable). Since a proof procedure concerns a logic the procedure takes a special form, superimposing a search strategy on an inference calculus. We will consider a certain collection of proof procedures in the light of an inference calculus format that abstracts the concept of logic programming. This formulation allows us to look beyond SLD-resolution, the proof procedure that underlies Prolog, to generalizations and extensions that retain an essence of logic programming structure. The inference structure used in the formulation of the logic programming concept and first realization, Prolog, evolved from the work done in the subdiscipline called automated theorem proving. While many proof procedures have been developed within this subdiscipline, some of which appear in Volume 1 of this handbook, we will present a narrow selection, namely the proof procedures which

are clearly ancestors of the first proof procedure associated with logic programming, SLD-resolution. Extensive treatment of proof procedures for automated theorem proving appear in Bibel [Bibel, 1982], Chang and Lee [Chang and Lee, 1973] and Loveland [Loveland, 1978]. Although the consideration of proof procedures for automated theorem proving began about 1958 we begin our overview with the introduction of the resolution proof procedure by Robinson in 1965. We then review the linear resolution procedures, model elimination and SL-resolution procedures. Our exclusion of other proof procedures from consideration here is due to our focus, not because other procedures are less important historically or for general use within automated or semi-automated theorem process. After a review of the general resolution proof procedure, we consider the linear refinement for resolution and then further restrict the procedure format to linear input resolution. Here we are no longer capable of treating full first-order logic, but have forced ourselves to address a smaller domain, in essence the renameable Horn clause formulas. By leaving the resolution format, indeed leaving traditional formula representation, we see there exists a linear input procedure for all of first-order logic.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Constraint Logic Programming: A Survey. Constraint Logic Programming: A Survey.

Constraint Logic Programming (CLP) began as a natural merger of two declarative paradigms: constraint solving and logic programming. This combination helps make CLP programs both expressive and flexible, and in some cases, more efficient than other kinds of programs. Though a relatively new field, CLP has progressed in several and quite different directions. In particular, the early fundamental concepts have been adapted to better serve in different areas of applications. In this survey of CLP, a primary goal is to give a systematic description of the major trends in terms of common fundamental concepts. Consider first an example program in order to identify some crucial CLP concepts. The program below defines the relation  $\text{sumto}(n, 1 + 2 + \dots + n)$  for natural numbers  $n$ . . . .  $\text{sumto}(0, 0)$ . . . .  $\text{sumto}(N, S) :- N \geq 1, N \leq S, \text{sumto}(N - 1, S - N)$ . . . The query  $S \leq 3, \text{sumto}(N, S)$  gives rise to three answers ( $N = 0, S = 0$ ), ( $N = 1, S = 1$ ), and ( $N = 2, S = 3$ ), and terminates.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5:

## Logic Programming. The Role of Abduction in Logic Programming. The Role of Abduction in Logic Programming.

This paper extends and updates our earlier survey and analysis of work on the extension of logic programming to perform abductive reasoning [Kakas et al., 1993]. The purpose of the paper is to provide a critical overview of some of the main research results, in order to develop a common framework for evaluating these results, to identify the main unresolved problems, and to indicate directions for future work. The emphasis is not on technical details but on relationships and common features of different approaches. Some of the main issues we will consider are the contributions that abduction can make to the problems of reasoning with incomplete or negative information, the evolution of knowledge, and the semantics of logic programming and its extensions. We also discuss recent work on the argumentation-theoretic interpretation of abduction, which was introduced in the earlier version of this paper. The philosopher Peirce first introduced the notion of abduction. In [Peirce, 1931-58] he identified three distinguished forms of reasoning. Deduction, an analytic process based on the application of general rules to particular cases, with the inference of a result. Induction, synthetic reasoning which infers the rule from the case and the result. Abduction, another form of synthetic inference, but of the case from a rule and a result. Peirce further characterised abduction as the “probational adoption of a hypothesis” as explanation for observed facts (results), according to known laws. “It is however a weak kind of inference, because we cannot say that we believe in the truth of the explanation, but only that it may be true” [Peirce, 1931-58]. Abduction is widely used in common-sense reasoning, for instance in diagnosis, to reason from effect to cause [Charniak and McDermott, 1985; Pople, 1973]. We consider here an example drawn from [Pearl, 1987]. Abduction consists in computing such explanations for observations. It is a form of non-monotonic reasoning, because explanations which are consistent with one state of a knowledge base may become inconsistent with new information. In the example above the explanation *rained-last-night* may turn out to be false, and the alternative explanation *sprinkler-was-on* may be the true cause for the given observation.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Transformation of Logic Programs. Transformation of Logic Programs.

Program transformation is a methodology for deriving correct and efficient programs from specifications. In this chapter, we will look at the so called 'rules + strategies' approach, and we will report on the main techniques which have been introduced in the literature for that approach, in the case of logic programs. We will also present some examples of program transformation, and we hope that through those examples the reader may acquire some familiarity with the techniques we will describe. The program transformation approach to the development of programs has been first advocated in the case of functional languages by Burstall and Darlington [1977]. In that seminal paper the authors give a comprehensive account of some basic transformation techniques which they had already presented in [Darlington, 1972; Burstall and Darlington, 1975]. Similar techniques were also developed in the case of logic languages by Clark and Sickel [1977], and Hogger [1981], who investigated the use of predicate logic as a language for both program specification and program derivation. In the transformation approach the task of writing a correct and efficient program is realized in two phases. The first phase consists in writing an initial, maybe inefficient, program whose correctness can easily be shown, and the second phase, possibly divided into various subphases, consists in transforming the initial program with the objective of deriving a new program which is more efficient. The separation of the correctness concern from the efficiency concern is one of the major advantages of the transformation methodology. Indeed, using this methodology one may avoid some difficulties often encountered in other approaches. One such difficulty, which may occur when following the stepwise refinement approach, is the design of the invariant assertions, which may be quite intricate, especially when developing very efficient programs. The experience gained during the past two decades or so shows that the methodology of program transformation is very valuable and attractive, in particular for the task of programming 'in the small', that is, for writing single modules of large software systems.

. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming.

Logic is now widely recognized as one of the foundational disciplines of computing and has applications in virtually all aspects of the subject, from software engineering

and hardware to programming languages and artificial intelligence. The Handbook of Logic in Artificial Intelligence and its companion The Handbook of Logic in Computer Science were created in response to the growing need for an in-depth survey of these applications. This handbook comprises five volumes, each an in-depth overview of one of the major topics in this area. The result of years of cooperative effort by internationally renowned researchers, it will be the standard reference work in AI for years to come. Volume 5 focuses on logic programming. The chapters, which in many cases are of monograph length and scope, emphasize possible unifying themes.

. Logic Programming. Algebra of Logic Programming. Logic Programming. Logic Programming with Requests. Logic Programming. The Association for Logic Programming. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Semantics for Disjunctive and Normal Disjunctive Logic Programs. Semantics for Disjunctive and Normal Disjunctive Logic Programs. During the past 20 years, logic programming has grown from a new discipline to a mature field. Logic programming is a direct outgrowth of work that started in automated theorem proving. The first programs based on logic were developed by Colmerauer and his students [Colmerauer et al., 1973] at the University of Marseilles in 1972 where the logic programming language PROLOG was developed. Kowalski [1974] published the first paper that formally described logic as a programming language in 1974. Alain Colmerauer and Robert Kowalski are considered the founders of the field of logic programming, van Emden and Kowalski [van Emden and Kowalski, 1976] laid down the theoretical foundation for logic programming. In the past decade the field has witnessed rapid progress with the publication of several theoretical results which have provided a strong foundation for logic programming and extended the scope of logic as a programming language. The objective of this article is to outline theoretical results that have been developed in the field of logic programming with particular emphasis to disjunctive logic programming. Disjunctive logic programming is an extension of logic programming and is useful in representing and reasoning with indefinite information. A disjunctive logic program consists of a finite set of implicitly quantified universal clauses of the form:  $\dots A_1, \dots, A_m \rightarrow B_1, \dots, B_n$   $m > 0$  and  $n \geq 0$  (1)  $\dots$  where the  $A_i$ 's and the  $B_j$ 's are atoms. The atoms in the left of the implication sign form a disjunction and is called the head of the formula

and those on the right form a conjunction and is called the body of the formula. The formula is read as “A1 or A2 or ... or Am if B1 and B2 and ... and Bn.” There are several forms of the formula that one usually distinguishes. If the body of the formula is empty, and the head is not, the formula is referred to as a fact. If both are not empty the formula is referred to as a procedure.

. Logic Programming. Logic Programming in Oz with Mozart. Logic Programming. Reactive Logic Programming by Reinforcement Learning. Handbook of Logic in Artificial Intelligence and Logic Programming: Volume 5: Logic Programming. Negation as Failure, Completion and Stratification. Negation as Failure, Completion and Stratification.

The usual way of introducing negation into Horn clause logic programming is by ‘negation as failure’: if A is a ground atom . . . the goal  $\neg A$  succeeds if A fails the goal  $\neg A$  fails if A succeeds. . . . This is obviously not classical negation, at least not relative to the given program P; the fact that A fails from P does not mean that you can prove  $\neg A$  from P, e.g. if P is . . . a ?  $\neg b$  . . . then ? - b fails so, using negation as failure, ? - a succeeds, but a is not a logical consequence of P. You could deal with classical negation by using a form of resolution which gave a complete proof procedure for full first order logic. To a logician this would be the natural thing to do. Two reasons are commonly given for why this is not done. The first is that it is believed by most, but not all, practitioners, that this would be infeasible because it would lead to a combinatorial explosion, whereas negation as failure does not, since it is not introducing any radically new methods of inference, just turning the old ones round. The second is that, in practical logic programming, negation as failure is often more useful than classical negation. This is the case when the program is a database, e.g. an airline timetable. You list all the flights there are. If there is no listed flight from Zurich to London at 12.31, then you conclude that there is no such flight. The implicit use of negation as failure here saves us the enormous labour of listing all the non-existent flights. This implicit usage is made precise in the closed world assumption, one of the two commonest declarative semantics given for negation as failure. This was introduced by Reiter [1978] and formalises the idea that the database contains all the positive information about objects in the domain, that any positive ground literal which is not implied by the program is assumed to be false.

. Logic Programming. Mutable Terms in a Tabled Logic Programming System. Logic Programming. Temporal Probabilistic Logic Programs. Theory and Practice of Logic Programming. Theory and Practice of Logic Programming. Programming in logic without logic programming.

In previous work, we proposed a logic-based framework in which computation is the execution of actions in an attempt to make reactive rules of the form  $\text{if antecedent then consequent}$  true in a canonical model of a logic program determined by an initial state, sequence of events, and the resulting sequence of subsequent states. In this model-theoretic semantics, reactive rules are the driving force, and logic programs play only a supporting role. In the canonical model, states, actions, and other events are represented with timestamps. But in the operational semantics (OS), for the sake of efficiency, timestamps are omitted and only the current state is maintained. State transitions are performed reactively by executing actions to make the consequents of rules true whenever the antecedents become true. This OS is sound, but incomplete. It cannot make reactive rules true by preventing their antecedents from becoming true, or by proactively making their consequents true before their antecedents become true. In this paper, we characterize the notion of reactive model, and prove that the OS can generate all and only such models. In order to focus on the main issues, we omit the logic programming component of the framework.

. Logic Programming. Conditional Logic Programming

*statistical mechanics problems and solutions pathria airbus a320 technical blank country fact sheet template gstore ore aisc manual of steel construction 13th edition pdf download organizational structure for oil and gas companies*

## STATISTICAL MECHANICS PROBLEMS AND SOLUTIONS PATHRIA

### **Statistical Mechanics Problems and Solutions: A Pathria Path**

**Introduction** Statistical mechanics is a branch of physics that applies statistical methods to the macroscopic properties of matter. It provides a powerful framework

for understanding the behavior of large systems of particles, from gases to liquids to solids. However, solving statistical mechanics problems can be challenging due to the complex relationships between the microscopic and macroscopic scales.

### **Question 1: Maxwell-Boltzmann Distribution**

- Problem: Derive the Maxwell-Boltzmann distribution for the speed of gas molecules using the assumption that the molecules are non-interacting.
- Solution: The Maxwell-Boltzmann distribution describes the probability of finding a gas molecule with a given speed at a given temperature. Using the principles of equal a priori probabilities and the conservation of energy, it can be shown that the distribution is given by:  $f(v) = \left( \frac{4\pi v^2}{\sqrt{kT}} \right) \exp(-mv^2 / 2kT)$

### **Question 2: Ideal Gas Law**

- Problem: Use statistical mechanics to derive the ideal gas law, which relates pressure, volume, temperature, and number of particles.
- Solution: The ideal gas law is a consequence of the kinetic energy of gas molecules and their interactions with the walls of a container. By considering the number of collisions between molecules and walls per unit time, it can be shown that the pressure exerted by the gas is proportional to the number of molecules, the temperature, and inversely proportional to the volume.

### **Question 3: Equipartition of Energy**

- Problem: Explain the concept of equipartition of energy in classical statistical mechanics and its implications for the specific heat capacity of gases.
- Solution: Equipartition of energy states that each quadratic term in the Hamiltonian contributes an average of  $1/2 kT$  to the energy of the system. For a gas of monatomic molecules, the Hamiltonian involves only translational energy, leading to a specific heat capacity of  $3R/2$ , where  $R$  is the universal gas constant.

### **Question 4: Partition Function**

- **Problem:** What is the partition function and how is it used in statistical mechanics to calculate thermodynamic properties?
- **Solution:** The partition function is a sum over all possible microstates of a system, weighted by their Boltzmann factors. It provides a complete description of the statistical behavior of the system and can be used to calculate thermodynamic properties, such as energy, entropy, and specific heat capacity, by performing appropriate derivatives.

**Conclusion** Solving statistical mechanics problems requires a deep understanding of the underlying principles and the ability to apply them to specific situations. Pathria's book "Statistical Mechanics" provides a comprehensive guide to the subject, offering numerous examples and solutions to help students master the material. By working through these problems, students can gain valuable insights into the microscopic foundations of macroscopic phenomena and develop their problem-solving skills in statistical mechanics.

## **AIRBUS A320 TECHNICAL**

**What is the technology of the A320?** A320 flight deck The aircraft is the first subsonic commercial aircraft to be fitted with fly-by-wire controls. Instead of having a central control column and aileron wheel, each pilot has a sidestick controller.

**What are the flight control systems of the A320?** As an example, the A320 has a total of seven flight control computers - two ELACs (Elevator Aileron Computer), three SECs (Spoilers Elevator Computer) and two FACs (Flight Augmentation Computers) - whereas the A330 has a total of five computers - three PRIMs (Flight Control Primary Computer) and two SECs (Flight ...

**What is the structure of the A320?**

**What is the Airbus A320 summary?** The Airbus A320 is a low-wing airliner with twin turbofans and a conventional tail. The Airbus A320 family are narrow-body (single-aisle) aircraft with a retractable tricycle landing gear and powered by two wing pylon-mounted turbofan engines.

**Which technology is used by Airbus?** Airbus actively uses advanced computing solutions in these areas. We strongly believe quantum computing, in tandem with more traditional high-performance computing (HPC) solutions, can help us to solve key computationally intensive tasks.

**What kind of engine is in the Airbus A320?** The Airbus A320 was the world's first series production aircraft with fly-by-wire and side-sticks controls. It is powered by either CFM International CFM56-5 or International Aero Engines (IAE) V2500 engines.

**Does A320 have TCAS?** The AP/FD TCAS function is an additional flight guidance mode of the autoflight system available for A320, A330, A350 and A380 aircraft.

**What are the three hydraulic systems of the A320?** The A320 hydraulic system is composed of three different and fully independent circuits: Green, Yellow & Blue. The users are shared between the systems in order to ensure the control of the aircraft, even when one system is inoperative.

**What are the 4 main flight controls?** Aircraft flight control systems consist of primary and secondary systems. The ailerons, elevator (or stabilator), and rudder constitute the primary control system and are required to control an aircraft safely during flight.

**How does A320 PTU work?** The PTU is a hydraulic motor pump which transfers hydraulic power between the green and yellow systems without transfer of fluid. It operates automatically, whenever the pressure differential between the two systems reaches 500 PSI.

**How does a A320 hydraulic system work?** An electric pump pressurizes the BLUE system. In case of emergency, a pump driven by a Ram Air Turbine (RAT) pressurizes this system. When AC electrical power is available and the electric pump pushbutton is at AUTO position, the BLUE system is automatically energized: In flight.

**What is the military version of the A320?** In July 2018, Airbus was evaluating an A320neo variant for ISR missions, particularly maritime patrol and anti-submarine warfare, and military transport. The aircraft will be able to take roll-on/roll-off mission

payloads to carry passengers, troops, VIPs, patients, or cargo.

**What is the Chinese version of the A320?** The Comac C919 is a narrow-body airliner developed by Chinese aircraft manufacturer Comac.

## **BLANK COUNTRY FACT SHEET TEMPLATE**

### **GSTOREORE**

**How to create a fact sheet template?**

**How do you produce a fact sheet?**

**What is a country fact file?** This template includes a range of some key facts about countries, such as population, capital city, and famous landmarks, giving children a broad basic knowledge of the country. There's also a section for other interesting facts, for any funny or strange facts children come across in their research!

**How to construct a fact sheet?**

**Does Word have a fact sheet template?** Use this Fact Sheet template (MS Word) to summarize the key facts in your product. This Fact Sheet template provides readers with a 1-2 page overview of a product, program, issue, event or other topic.

**What does a good fact sheet look like?** Clear, concise language paired with straightforward graphics does the trick. Make sure your design isn't cluttered. A clean layout with plenty of white space helps key points stand out. Stick to your brand's colors, fonts, and logos for a cohesive look.

**What is a basic fact sheet?** A fact sheet is a short, typed or hand-written document that contains the most relevant information about a particular subject in the least amount of space. The goal is to provide facts and key points about a topic in a clear, concise, and easy-to-understand way.

**How do I create a fact sheet in Google?**

**What should a fact file look like?** To create a fact file, your child will need to start with a clear title, highlight an interesting fact at the top, use bullet points for clarity,

and consider adding pictures for engagement. They should include a "quick facts" section for memorable information.

### **Where can I get information about a country?**

**What is the format of fact file?** Title: tells the reader what the factfile is about  
Introductory paragraph: a summary of what the fact file is about. Pictures and diagrams: gives more information to the reader about the piece of text. Captions: description underneath a picture or diagram to describe what it is.

**What is a state fact sheet?** The ERS State Fact Sheets provide information on population, income, poverty, food security, education, employment/unemployment, farm characteristics, farm financial indicators, top commodities, and agricultural exports.

**Where can I make a fact sheet?** Create an eye-catching fact sheet with Piktochart's customization features to suit your specific needs. Whether you're creating a health fact sheet, a product fact sheet, or anything in between, our template has got you covered.

**What is a fact sheet PDF?** What is it. A Fact Sheet is a document that provides clear and easy to read information on a topic. Often, but not always, the purpose of the fact sheet is to convince the reader to do something, like hand washing often to avoid viruses, or joining an organization that saves the spotted owl, etc.

### **How to make a fact sheet in PowerPoint?**

**How to do a fact sheet template?** A fact sheet should preferably be a single page long. The information mentioned in the fact sheet should be brief and concise. Since visual information is easy to grasp, a fact sheet should be visually appealing comprising tables, charts, graphs, and bullet points. Fact sheets should be easily readable.

**Where can I find free Word templates?** You can download free, pre-built document templates with useful and creative themes from Office when you click File > New in your Office app. Templates can include calendars, business cards, letters, cards, brochures, newsletters, resumes, cover letters, presentations, social media and

much more.

**How long should a fact sheet be?** Fact sheets are typically one-page documents that emphasize the abbreviated arguments of an issue (1). Space is typically very limited, so you should focus on only the most compelling points.

**What is a key fact sheet?** The KFS is a one-page document that contains information such as the rate of interest, loan amount, loan term, processing fees, documentation charges, foreclosure charges and late payment fee, among others.

**What is a legal fact sheet?** Fact Sheet means a document that briefly sets forth the principal facts and the significant factual, legal, methodological, and policy questions considered in preparing a draft decision.

**What is the difference between a fact sheet and an infographic?** A fact sheet is usually (mostly) text. It relies on the reader understanding the language it is written in. An infographic is heavily image based, & while it does have text as well, a well made infographic can be understood without the reader understanding the language the text is written in.

**What is a basic example of fact?** Fact – Facts are useful to inform or make an argument. Examples: The United States was established in 1776. The pH levels in acids are lower than the pH levels in alkalines.

**What is a fact file example?** A fact file is a collection of the most important information on a particular subject or thing. For example, when filling in our fact file template on a person you could include: Their name. Their date of birth.

**What information should be included on a personal fact sheet?** In your fact sheet, you can include information regarding your assets and liabilities, property information, insurance policies, Advance Health Care Directives, and Wills.

**Does Google Sheets have templates?** Use a Google template On your computer, go to Google Docs, Sheets, Slides, or Forms. At the top right , click Template Gallery. Click the template you want to use. A copy of the template opens.

**How do I create a free Google Sheet?**

**Is Google Sheets free?** Google Sheets is a web-based application that provides users access to seamless collaboration or solo creation, editing, and management of spreadsheets from their browser. The spreadsheet solution, which launched in 2006 and is part of Google's G Suite, is free and easy to use.

**How to create a fact sheet in PowerPoint?**

**How do I create a fact sheet in Google?**

**How do I create a worksheet template in Word?**

**How do you write a fact file format?** To create a fact file, your child will need to start with a clear title, highlight an interesting fact at the top, use bullet points for clarity, and consider adding pictures for engagement. They should include a "quick facts" section for memorable information.

**Where can I find templates in PowerPoint?** When you first open PowerPoint, in the left column, select New or, if you already have a PowerPoint presentation open, go to File > New. In the New window, open the templates folder with your organization's name on it. You'll see branded templates within or folders containing branded templates.

**How do I download a template for PowerPoint?**

**Can you create a template in PowerPoint?** Create a PowerPoint template On the Design tab, select Slide Size > Custom Slide Size and choose the page orientation and dimensions you want. On the View tab, in the Master Views group, choose Slide Master. The slide master is the largest slide image at the top of the slide thumbnail list, to the left of your slides.

**Where can I make a fact sheet?** Create an eye-catching fact sheet with Piktochart's customization features to suit your specific needs. Whether you're creating a health fact sheet, a product fact sheet, or anything in between, our template has got you covered.

**What should be included in a factsheet?** A fact sheet lists all the key information, facts, and figures around a particular topic, in a visual manner, with the help of files, charts, images, etc. Fact sheets often contain company overview, product

information, statistics, technical data, FAQs, lists, “how-to” pages, educational material, and so on.

**Does Google Sheets have templates?** Use a Google template On your computer, go to Google Docs, Sheets, Slides, or Forms. At the top right , click Template Gallery. Click the template you want to use. A copy of the template opens.

**How do I create a template spreadsheet?** Open the workbook you want to use as a template. Click File > Export. Under Export, click Change File Type. In the Workbook File Types box, double-click Template.

**How to make a fillable worksheet in Word?**

**What is a template worksheet form?** An Excel template is a predesigned sheet that can be used to create new worksheets with the same layout, formatting and formulas. With templates, you don't need to recreate the basic elements every time as they are already integrated into the spreadsheet.

**What to include in a fact file about a country?** The prompts on the worksheet include the capital city, population, currency, language, and neighbouring countries. There are also prompts to encourage your students to write about famous people, famous landmarks, and fun facts.

**What does a good fact file look like?** Title: tells the reader what the factfile is about  
Introductory paragraph: a summary of what the fact file is about.  
Pictures and diagrams: gives more information to the reader about the piece of text.  
Captions: description underneath a picture or diagram to describe what it is.

**How to make a fact file in Google Docs?**

## **AISC MANUAL OF STEEL CONSTRUCTION 13TH EDITION PDF DOWNLOAD**

**What is the latest AISC steel construction manual?** Since the early 2000s, the Manual has been updated and reissued as a new edition every five to six years to keep up with developments in structural steel codes and standards and to

incorporate new materials. The most recent edition is the 16th, published in 2023.

**What is the latest edition of AISC code?** Code Update: The 2022 AISC Specification for Structural Steel Buildings. The latest edition of the American Institute of Steel Construction (AISC) flagship standard, the “2022 Specification for Structural Steel Buildings (ANSI/AISC 360)” (the Specification), has hit the streets (get it at [aisc.org/2022spec](https://aisc.org/2022spec)).

**What is the AISC specification A3 1c?** AISC Specification A3. 1c uses this term to describe rolled shapes with flange thicknesses exceeding 2 in., and built-up heavy shapes composed of components made from plate exceeding the same dimension.

**What is the AISC Specification section J3 1?** The AISC Specification Section J3. 1 indicates that the following conditions require slip-critical joints: As required by the RCSC Specification. The extended portion of bolted, partial-length cover plates, as required in Section F13.

**When did the AISC 14th edition come out?** The 14th edition, released in 2011, contains several updates and revisions from the 13th edition, including the new HP18 and HP16 series, updated connection tables based on increased bolt shear strength values, revised single-plate and extended single-plate connection design procedures, enhanced prying action procedure, ...

**What is the difference between AISC and ASCE?** ASCE 7 gives requirements for the determination of loads and load combinations, and limits on the resulting drifts and stability. AISC 360 gives requirements for the analysis and design of steel structures.

**What is new in the AISC 16th edition?** The 16th edition features: New 50 ksi steel design tables. Properties and dimensions for 210 new HSS shapes. Revised and expanded discussion of prying action.

**What is the latest ASME Code Edition?** The 2023 code edition for rated flanges now points to the 2020 edition for those requirements, specifically for ASME B16. 5 and B16. 47 like Tables U-3 under Division 1 and Table 1.1 under Division 2. Additionally, the 2023 editions of ASME Div 1 and Div 2 now point to the 2022 edition of ASCE 7 rather than 2016.

**What is the AISC standard?** The AISC Code of Standard Practice provides a framework for a common understanding of the acceptable standards when contracting for structural steel. As such, it is useful for everyone associated with construction in structural steel.

**What's new in the AISC 16th edition?** New features include new 50-KSI steel design tables; properties and dimensions for 210 new HSS shapes; revised and expanded discussion of prying action; new and revamped tables for design of double-angle connections, single-plate connections, single-angle connections, and shear end-plate connections; new section on the ...

**When did the 15th edition of steel Manual come out?** The 15th Edition Steel Construction Manual was first released in 2017.

**What is the latest code for steel structure design?** The IS code for steel is 800 which lays down the general guidelines for civil engineers and builders to follow during construction in steel. IS: 800 however is not the only IS code to be followed.

**What is AISC 207?** Standard for Certification Programs (AISC 207-23) The revised AISC 207-23 incorporates updates across several key areas, including the glossary, welding specifications, inspection and testing procedures, as well as providing enhanced clarity regarding subcontracting and record retention.

## **ORGANIZATIONAL STRUCTURE FOR OIL AND GAS COMPANIES**

**What is the business structure of oil and gas?** The industry is often divided into three segments: upstream, the business of oil and gas exploration and production; midstream, transportation and storage; and. downstream, which includes refining and marketing.

**What type of market structure is the oil and gas industry?** Oligopolies are different from monopolies, in which only one firm is the dominant producer in the industry. Examples of oligopolies can be found across major industries like oil and gas, airlines, mass media, automobiles, and telecom.

**What is the structure of the petroleum industry?** The industry is usually divided into three major components: upstream, midstream, and downstream. Upstream regards exploration and extraction of crude oil, midstream encompasses transportation and storage of crude, and downstream concerns refining crude oil into various end products.

**What is the role of oil and gas companies?** Upstream, or exploration and production (E&P) companies, find reservoirs and drill oil and gas wells. Midstream companies are responsible for transportation from the wells to refineries. Downstream companies are responsible for refining and the sale of the finished products.

**How was the Standard oil company organized?** In 1881 Rockefeller and his associates placed the stock of Standard of Ohio and its affiliates in other states under the control of a board of nine trustees, with Rockefeller at the head. They thus established the first major U.S. "trust" and set a pattern of organization for other monopolies.

**What are the 3 industry segments of the oil and gas industry?**

**What type of business is an oil company?** Oil companies engage in exploration and production of oil ("upstream" activities), oil transportation and refining ("midstream"), and petroleum product wholesale and retail distribution ("downstream").

**Which of the following are the major segments of oil and gas industry?** The oil and gas industry is frequently divided into three segments: upstream, midstream and downstream.

**What type of market structure is ExxonMobil?** The market structure of ExxonMobil is an oligopoly.

**What is structural in oil and gas?** Structural Engineering in Oil & Gas is a branch of civil engineering that is concerned with the structural design of man-made structures like Equipment rack, Pipe Rack.

**What are the four segments of petroleum industry?**

**What is the structure of gas oil?** Petroleum has an average composition of 85% C, 13% H, and 2% N, S, and O. In natural gas, C is slightly lower and H higher. Distillation separates petroleum into fractions of increasingly higher complexity. HC types include paraffins, naphthenes, aromatics and olefins.

**Who is the industry leader in oil and gas?** ExxonMobil manages an industry-leading portfolio of resources and is one of the largest integrated fuels, lubricants and chemical companies in the world. It is also the largest refiner and marketer of petroleum products, with its chemical company being one of the largest globally.

**What is the largest oil company in the US?** ExxonMobil is the largest United States-based oil and gas producing company.

**How to do an oil and gas business?**

**What company controlled 90% of all oil business in the US?** In 1882, Standard Oil Trust created a network of Standard Oil companies throughout the country, led by a board of trustees, where Rockefeller owned over one third of the certificates. By the late 1880s, Standard Oil controlled 90% of American refineries.

**Do Rockefellers still own oil companies?** Heirs to the oil fortune created by John D. Rockefeller, who founded Standard Oil in 1870, are exiting the family business.

**Are the Rockefellers still rich?** Now entering its seventh generation with as many as 170 heirs, the Rockefeller family has maintained substantial wealth — they had an \$11 billion fortune in 2016, according to Forbes.

**What is the market structure of the oil and gas industry?** Based on type, the market is categorized into upstream, downstream, and midstream. The downstream segment captured the largest oil and gas market share of 49.13% in 2023, largely attributed to its critical role in refining, processing, and distributing oil and gas products.

**What are the four verticals of oil and gas?**

**What are the three stages of the oil and gas industry?** The three sectors of the oil and gas industry inherently interconnect. The upstream sector provides the raw

materials, the midstream sector transports these materials to the downstream sector, and the downstream sector refines and processes them into various end-user products.

**Who is the richest oil company?**

**Who are the big 6 oil companies?** In the maritime industry, a group of six companies that control the chartering of the majority of oil tankers worldwide are together referred to as "oil majors". These are: Shell, BP, ExxonMobil, Chevron, TotalEnergies and ConocoPhillips.

**What is the strongest oil and gas brand?** Brand Finance highlighted in its release that Petronas was ranked as the oil and gas sector's strongest brand, "retaining its AAA brand strength rating despite facing significant global challenges".

**What is E&P in oil and gas?** Exploration & production (E&P) is a specific sector within the oil and gas industry linked to the early stage of energy production, which generally involves searching for and extracting oil and gas. An E&P company finds and extracts the raw materials used in the energy business.

**What are the categories of procurement in the oil and gas industry?**

**What is the profit margin for the oil and gas industry?** Oil and gas production profit margins are volatile, varying widely with energy prices. The average net profit margin for oil and gas production was 4.7% in 2021 and 31.3% in Q4 2021.

**What organizational structure does ExxonMobil use?** Exxon Mobil also has a decentralized structure, which allows for quick decision-making and fast execution. The company is divided into several business segments, each with their own leadership team.

**What market structure are oil industries best examples of?** The oil industry is largely oligopolistic, meaning a small number of firms hold a significant market share. This can influence pricing as these large producers have the power to alter supply levels and, consequently, prices.

**What is ExxonMobil's business model?** As previously highlighted, ExxonMobil's business model is based on the exploration, extraction, refining, and sale of oil and

gas products, as well as the development and deployment of new technologies to improve energy efficiency and reduce environmental impact.

**What is the structure of gas oil?** Petroleum has an average composition of 85% C, 13% H, and 2% N, S, and O. In natural gas, C is slightly lower and H higher. Distillation separates petroleum into fractions of increasingly higher complexity. HC types include paraffins, naphthenes, aromatics and olefins.

**What is structural in oil and gas?** Structural Engineering in Oil & Gas is a branch of civil engineering that is concerned with the structural design of man-made structures like Equipment rack, Pipe Rack.

**What is the basic structure of oil?** Fats and oils are composed of molecules known as triglycerides, which are esters composed of three fatty acid units linked to glycerol. An increase in the percentage of shorter-chain fatty acids and/or unsaturated fatty acids lowers the melting point of a fat or oil.

**What are the oil structures called?** Offshore oil and gas structures are commonly referred to as "rigs" or "platforms." These terms actually refer to the drilling rigs used to drill wells and the large structures that serve as production platforms at producing well sites.

**What is the simple structure of oil?** Structures of Fats and Oils Fats and oils are called triglycerides (or triacylglycerols) because they are composed of three fatty acid units joined to glycerol: If all three OH groups on the glycerol molecule are replaced with the same fatty acid, the resulting ester is called a simple triglyceride.

**What is the jacket structure of oil and gas?** Jacket structures are three-dimensional space frames made up of large tubular steel members. Typically, the jacket takes the loadings from the topside and the sea environment, and is anchored to the seabed using steel piles.

**What does GOR stand for in oil and gas?** The proportion between oil and natural gas obtained from a producing well under given conditions of temperature and pressure. Commonly abbreviated GOR and measured in standard cubic feet of gas per barrel of oil (SCF/BBL).

**What is WBS in oil and gas?** Work Breakdown Structure (WBS) | OGDE - Oil & Gas Development Estimating Ltd.

**What is the market structure of the oil and gas industry?** Based on type, the market is categorized into upstream, downstream, and midstream. The downstream segment captured the largest oil and gas market share of 49.13% in 2023, largely attributed to its critical role in refining, processing, and distributing oil and gas products.

**What are the functional area of the oil and gas industry?** Each industry will have its own operating expenses for example: A major Oil & Gas Upstream company might have the functional area in the form of - Production Expense, Gas Plant, Exploration, Seismic, Dry Hole, Terminals & Rail, Abandonment, Drilling, Corrective Maintenance, Preventive Maintenance, Refurbishment, ...

**What are the 3 basic groups of oil?** There are three types of base oils: mineral, vegetable, and synthetic. Mineral oil comes from crude oil and the quality depends on the refining process. Lubes made from vegetable oils are called biolubricants. Synthetic oils are man-made fluids and can be beneficial for use in extreme conditions.

**What is the general structure of petroleum?** Petroleum or crude oil is a naturally occurring, flammable, complex and variable mixture of hydrocarbons plus other organic compounds in the liquid state. Petroleum contains 83- 87% carbon, 10 to 14% hydrogen, with trace amounts of nitrogen, oxygen and sulfur. The sulfur content of oil can be as high as 6%.

**What is the structure of an oil refinery?** Oil refineries are typically large, sprawling industrial complexes with extensive piping running throughout, carrying streams of fluids between large chemical processing units, such as distillation columns. In many ways, oil refineries use many different technologies and can be thought of as types of chemical plants.

**What is a platform in oil and gas?** An oil platform (also called an oil rig, offshore platform, oil production platform, etc.) is a large structure with facilities to extract and process petroleum and natural gas that lie in rock formations beneath the seabed.

**What organization controls the most oil?**

**What is the biggest oil group?** Also the third largest energy company generally worldwide according to Forbes' 2024 Global 2000 list, Saudi Aramco is a state-owned oil and gas company based in Saudi Arabia. It is the biggest oil and gas company in the world and is involved in the exploration, production, refining and marketing of oil and natural gas.